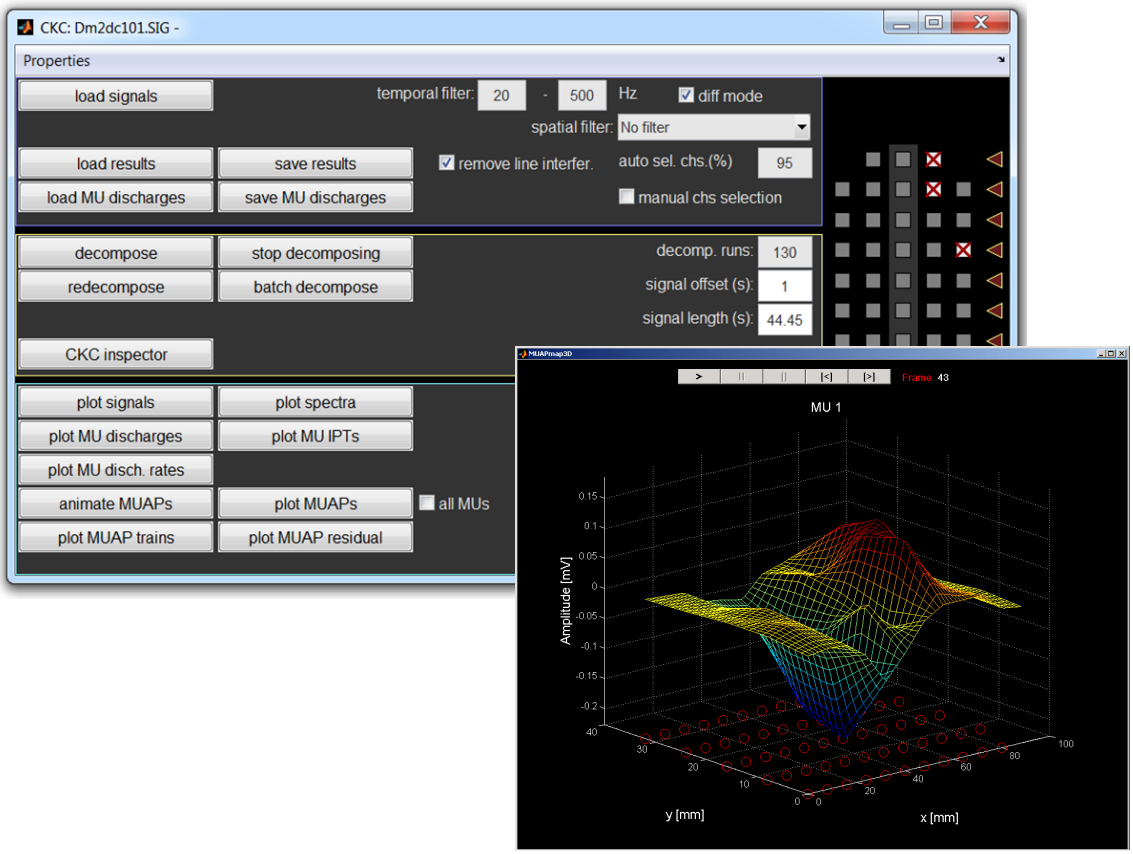


DEMUSE tool – user manual

Version 4.0

24/02/2014

CONTENTS

1 Introduction 4
2 DEMUSE tool components 4
3 DEMUSE tool installation 5
3.1 Requirements 5
3.2 DEMUSE files and folders 6
3.2.1 Automatic Installation (Windows OS and Unbuntu only) 6
3.2.2 Manual Installation: Windows operating system 7
3.2.3 Manual installation on linux (Ubuntu) 8
3.2.4 Setting the Matlab path 9
4 Using DEMUSE tool 10
4.1 Starting DEMUSE tool 10
4.2 Loading EMG signals 11
4.3 Filtering EMG signals 13
4.4 Visualization of EMG signals 15
4.5 Decomposition of EMG signals 18
4.6 Editing and inspecting the decomposition results: CKC inspector 21
4.6.1 Adding and deleting MU discharges 23
4.6.2 Estimation of MUAP templates 25
4.6.3 Improvement of decomposition results 26
4.6.4 MU tracking 27
4.6.5 Managing the identified MUs 28
4.6.6 Assessing the accuracy of decomposition 29
4.6.7 Closing the CKC inspector window 30
4.7 Deleting the identified MUs 31
4.8 Plotting the decomposition results 31
4.8.1 MU discharge patterns plots 32
4.8.2 Smoothed discharge rate plots 33
4.8.3 Multichannel MUAP plots 34
4.8.4 3D MUAP map animation 36
4.8.5 Plots of reconstructed MUAP trains 37
4.9 Saving and reloading the decomposition results 41
5 Appendix I: CKCreader 43
Citations and technical support 45
References 46

1 Introduction

DEMUSE® is a Matlab program for visualization and decomposition of multichannel surface electromyograms (EMG). It runs on a standard PC and enables the user to:

- load and visualize the multichannel surface electromyograms;
- decompose surface EMG signals into contributions of individual motor units (MUs);
- inspect and edit the results obtained by automatic decomposition;
- display graphs of decomposition results, including plots of the MU discharge patterns, instantaneous discharge rate, motor unit action potentials (MUAPs) and their 3D animations;
- compare the original surface EMG signals to the reconstructed MU action potential (MUAP) trains;
- save and reload the decomposition results.

All the graphs are displayed as regular Matlab figures and can be freely manipulated by standard Matlab graphic tools (i.e., figure resizing, zooming, rotating, printing, etc.). User is referred to Matlab documentation for further details on the use of Matlab graphic user interface.

Note: *The current version of the DEMUSE supports decomposition of isometric surface EMG signals only, i.e., the signals, acquired during isometric muscle contractions. Intensive work on decomposition of dynamic surface EMG signals is currently in progress and support for dynamic conditions will be built into future versions of this decomposition tool.*

2 DEMUSE tool components

DEMUSE represents the third layer in three-tier system architecture (**Figure 1**). The first two layers comprise the electromyographic signal amplifier and the surface EMG acquisition software, respectively. The selection of amplifier and acquisition tool is left to the end user and is not described in this manual. In general, any amplifier for high-density EMG can be used. However the following list of hardware has been tested and is fully compatible with DEMUSE tool:

- Portable multi-channel acquisition system (LISiN, Politecnico di Torino, Italy)
- EMG-USB2 amplifier (OT Bioelettronica, Torino, Italy);

Acquired EMG signals are loaded into the DEMUSE tool by means of open source readers (see Section 5) and processed off-line (**Figure 1**).

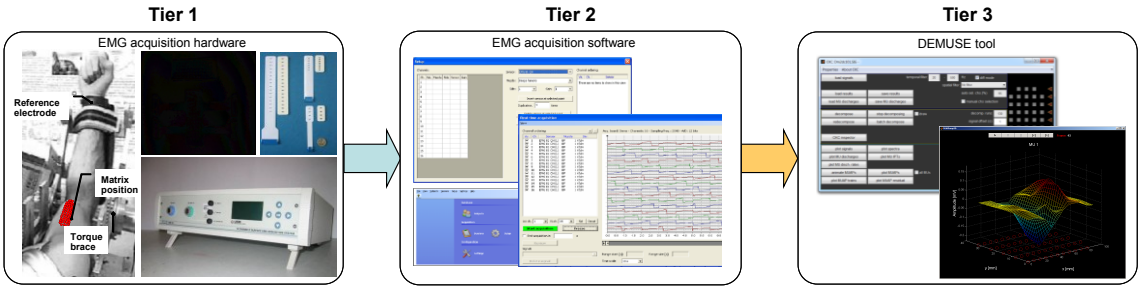


Figure 1: Three-tier architecture (with indicated data flows) of the surface EMG decomposition.

3 DEMUSE tool installation

DEMUSE tool runs in the Matlab environment. As such, it is supported by several personal computer (PC) platforms, including Windows, Linux and Mac OS. Beside the USB dongle supplied in the installation kit, it does not require any special hardware configuration. However, in the case of large number of channels and long EMG signals, a sufficient amount of RAM (8 GB or more) should be installed on the system in order to prevent extensive swapping of the memory space.

3.1 Requirements

Minimal hardware configuration:

- 1 GHz CPU;
- 20 MB disk;
- 8 GB RAM.

Recommended hardware configuration:

- 2 GHz CPU or higher;
- 100 Mb disk;
- 2 MB L2 cache or more;
- 32 GB RAM.

To run DEMUSE tool the following software should be properly installed:

- Matlab (http://www.mathworks.com/), version 7.3 or higher.
- SmartDongle driver - see Installation instructions bellow (http://www.smartdongle.com/DevelopmentCenter.aspx).

3.2 DEMUSE files and folders

DEMUSE tool comprises several Matlab's content-obscured executable files which are located in the directory

```
..\DEMUSE\programs\
```

Program documentation is located in the directory

```
..\DEMUSE\documentation\
```

3.2.1 Automatic Installation (Windows OS and Unbuntu only)

To install DEMUSE tool, it is required to run Matlab with administrator rights (in Windows use `run as Administrator` command, in Ubuntu use `sudo` command).

First, unzip installation archive. Run Matlab with administration rights and execute "`installGUI`" script:

```
>> installGUI
```

Dialog window with two textboxes appears (Figure 2). The upper textbox is used to insert the DEMUSE license number. The lower textbox is used to select the DEMUSE tool installation folder. The installation path can be inserted as a text or selected by pressing the `Browse` button. When `DEMUSE install folder` textbox is left blank, default installation path is used. In Windows operating system, the default installation path is set to the hard disk's partition where Matlab is installed. For example, when Matlab is installed in "`C:\Program Files\Matlab`" the default installation folder for DEMUSEtool is set to "`C:\DEMUSE`".

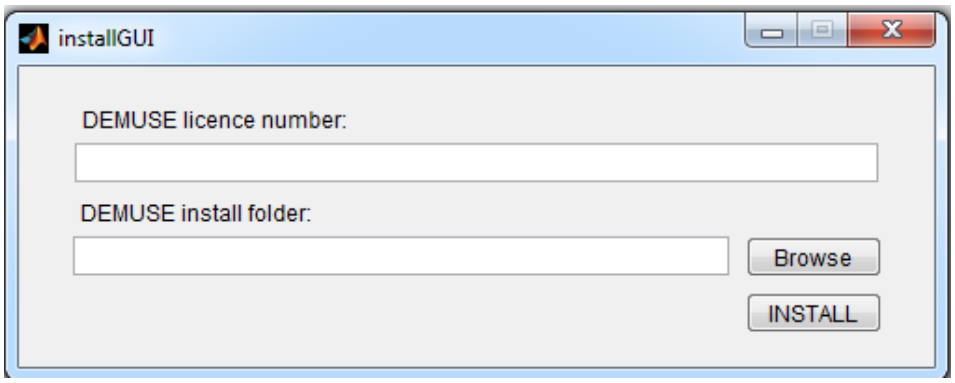


Figure 2: DEMUSE Installation dialog window.

After the successful installation, copy the `serial.mat` file to the DEMUSE installation folder.

Note: To properly install DEMUSE tool with install script, Matlab has to be run with administration rights. Automatic install was tested on different windows platforms and Ubuntu 13.10 (64-bit). 32bit Ubuntu is not supported at the moment.

3.2.2 Manual Installation: Windows operating system

DEMUSE tool can also be installed manually by performing the following steps:

Step 1: Download and extract the DEMUSE tool installation archive. Go to subfolder `drivers` and run "`SmartDongle_x64…`" (on 64 bit operating system) or "`SmartDongle_x86…`" executable (on 32 bit operating system). Afterwards, execute the "`vcredist_x64`" (64 bit operating system) or "`vcredist_x86`" executable (32 bit operating system).

Step 2: Copy the `DEMUSE` folder to the folder where the DEMUSE tool is to be installed.

Step 3: Copy files from `/dll/x64` (64 bit operating system) or `/dll/x86` folder (32 bit operating system) to the same local folder as in Step 2.

Step 4: Copy all the files and subfolders from `support\x64` (64 bit operating system) or `support\x86` (32 bit operating system) folder to the same local destination as in Step 2.

Step 5: In Windows Registry (e.g. by running the `regedit` command), create new key named `DEMUSE` under the `HKEY_LOCAL_MACHINE\SOFTWARE`. Add two string values. First string should be named `licence` and should contain the DEMUSE licence number (as purchased from the supplier). The second should be named `DEMUSEpath` and should contain absolute path to the DEMUSE tool installation folder (as set in Step 2).

Step 6: Copy `serial.mat` to the DEMUSE tool installation folder (as set in Step 2).

3.2.3 Manual installation on Linux (Ubuntu)

At the moment only 64 bit Ubuntu operating system is supported. DEMUSE tool can also be installed on other 64 bit Linux platforms, but the installation procedure could differ from the one described in the sequel. Please, contact the supplier for additional info and support.

DEMUSE tool can also be installed manually by performing the following steps:

Step 1: Unzip and copy the `DEMUSE` folder to the folder where the DEMUSE tool is to be installed. For example:

```
Ubuntu:$ unzip DEMUSE.zip
Ubuntu:$ cp ./DEMUSE /home/user/DEMUSE
```

Step 2: Copy all the files from `./so/` to the same local folder as in Step 1.

```
Ubuntu:$ cp ./so/ /home/user/DEMUSE
```

Step 3: Copy all the files and subfolders from `support/x86` to the same local folder as in Step 1.

```
Ubutnu:$ cp ./support/x86/ /home/user/DEMUSE
```

Step 4: Create the file `licence.dat` in the folder `/etc/DEMUSE/`. Open the `licence.dat` with text editor and copy the licence into the first row. In the second row, write absolute path to the installation folder chosen in the Step 1. Close and save the file.

```
Ubutnu:$ sudo gedit /etc/DEMUSE/licence.dat
```

Step 5: Copy `libstdc++.so.6.0.17` file from the `./so/` folder to `/sys/os/glnxa64` subfolder in Matlab root folder. This can be done in Matlab by executing the following command:

```
copyfile('./so/libstdc++.so.6.0.17', [matlabroot
'/sys/os/glnxa64/'], 'f');)
```

or in Linux terminal window:

```
Ubuntu:$ sudo cp ./so/libstdc++.so.6.0.17
"matlabroot"/sys/os/glnxa64
```

Step 6: Create symbolic link to the file from Step 5 (use Matlab's command:

```
unix(['ln -sf ' [matlabroot
'/sys/os/glnxa64/libstdc++.so.6.0.17 ']
[matlabroot /sys/os/glnxa64/libstdc++.so.6']]);).
```

or in Linux terminal window:

```
Ubuntu:$ sudo ln -sf
"matlabroot"/sys/os/glnxa64/libstdc++.so.6.0.17
"matlabroot"/sys/os/glnxa64/libstdc++.so.6
```

Note: To install DEMUSE tool, it is required to run Matlab with administrator rights (e.g., by using the `sudo` command in Ubuntu). The installation process creates several subfolders and files in the DEMUSE installation folder (see instructions above). These subfolders and files are created with root privileges and will not be accessible to standard (non-root) users. You must manually add the permissions in order to grant the access to these files also to non-root users. For example, to add all permissions to every user on the system, use the following command in Ubuntu terminal window:

```
Ubutnu:$ sudo chmod –R 777 /home/user/DEMUSE
```

3.2.4 Setting the Matlab path

To run the DEMUSE tool in Matlab, one must browse to the `DEMUSE\programs\` folder and run the `DEMUSE` command (see Section 4.1). Alternatively, you may set the path in Matlab environment to `DEMUSE\programs\` subfolder and run the DEMUSE tool from any location in matlab.

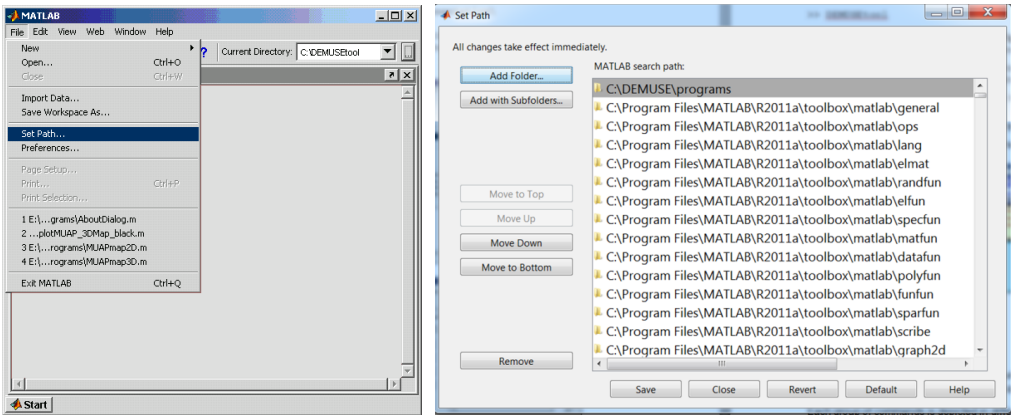


Figure 3: Setting the path in Matlab programme environment (in this case, DEMUSE tool was copied to the directory `c:\DEMUSE`).

4 Using DEMUSE tool

4.1 Starting DEMUSE tool

To start the DEMUSE tool, type the following command to Matlab command window:

```
>> DEMUSE
```

The main DEMUSE tool window appears (**Figure 4**). This window comprises four frames with the following groups of commands for:

- loading, band-pass filtering and visualization of the acquired surface EMG signals, and saving and reloading of the decomposition results;
- decomposition of acquired surface EMG signals and manual inspection of decomposition results;
- graphical plots and animations of the decomposition results;
- selection of channels, rows and columns in the matrix of surface EMG electrodes.

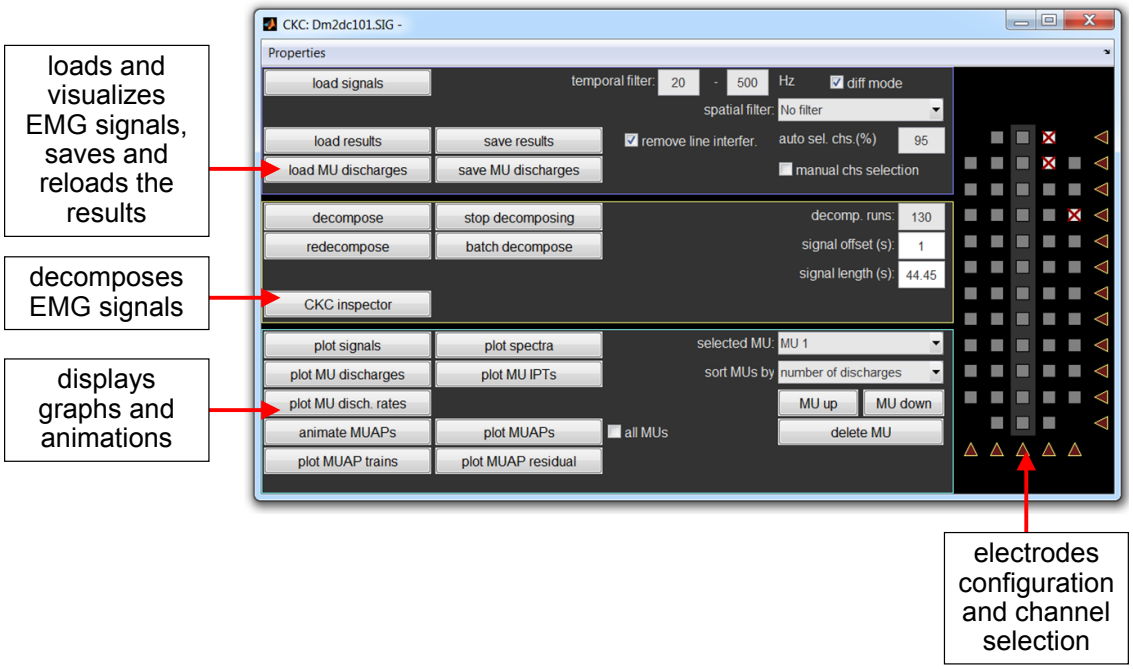


Figure 4: The main DEMUSE tool window with explanations of different command groups.

4.2 Loading EMG signals

To load the acquired surface EMG into DEMUSE tool click on "`load signals`" button (**Figure 5**, left panel). "`Load signals`" dialog window appears (**Figure 5**, right panel). Chose the folder and the file and click on "`Open`" button.

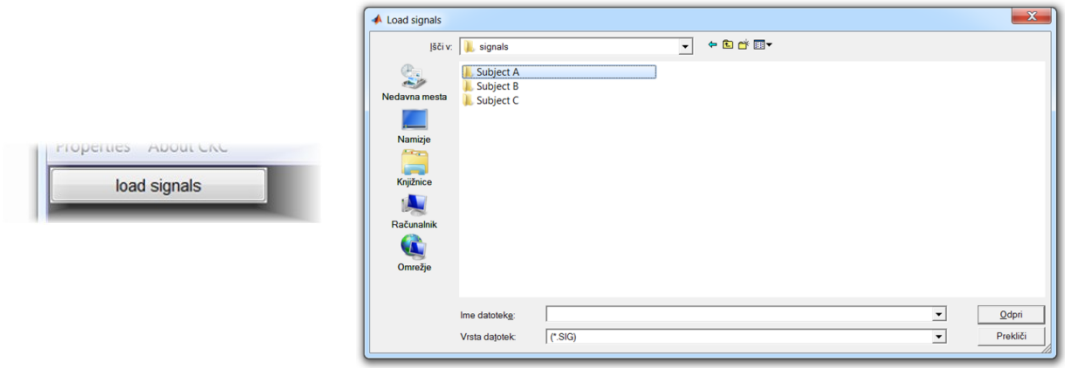


Figure 5: "`load signals`" button (left panel) and "`Load signals`" dialog window (right panel).

DEMUSE tool automatically filters the EMG files by their extensions. The list of extensions can be defined by selecting the `EMG file extensions` in the `Properties` menu. A dialog window appears (**Figure 6**). New file extension can be added by pressing the `add` button. To delete the existing file extensions, select the extension in the `File extension` list and press the `delete` button. The list of file extensions is automatically saved when the dialog window is closed.

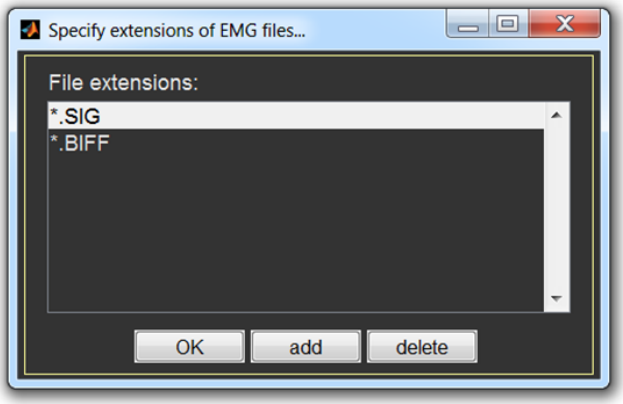


Figure 6: dialog window for definition of supported EMG file extensions.

DEMUSE tool supports arbitrary configurations and numbers of surface electrodes and stores the information about the acquisition modalities in so called readers (Section 5). This includes the sampling frequency, dimensions of the acquisition system, electrode configurations, etc. After closing "`Load signals`" dialog window, DEMUSE tool prompts for selection of a proper reader for EMG files (**Figure 7**). There are several

readers already implemented, supporting main acquisition systems currently provided by LISiN laboratory (Politecnico di Torino, Italy) and OT Bioelettronica (Torino, Italy). For more complex acquisition configurations or other acquisition systems, specialized reader of the EMG files can be implemented and added to the `Readers` directory (see Appendix I). DEMUSE tool automatically loads all the readers from the `Readers` directory and displays their descriptions in the `Reader Dialog Window` (**Figure 7**). To select the specific reader, click on the line with its name. Corresponding reader description is automatically displayed in the `Reader Description Panel`. To confirm the reader selection press `OK` button.

Proper reader can also be specified in a text file called `CKC_reader.txt` (see Appendix I). Simply write the name of corresponding reader in `CKC_reader.txt` file e.g.

```
CKC_reader_my_reader.m
```

and copy the file into the directory with corresponding EMG files. DEMUSE tool will automatically use the specified reader for all the EMG files in the corresponding directory.

After the signals are loaded, DEMUSE tool displays the number of acquired EMG channels and their relative spatial configuration in the `Channel selection frame` (**Figure 8**).

In the case of loading failure, the Error Dialog Window will appear (**Figure 9**) with a short description of the error.

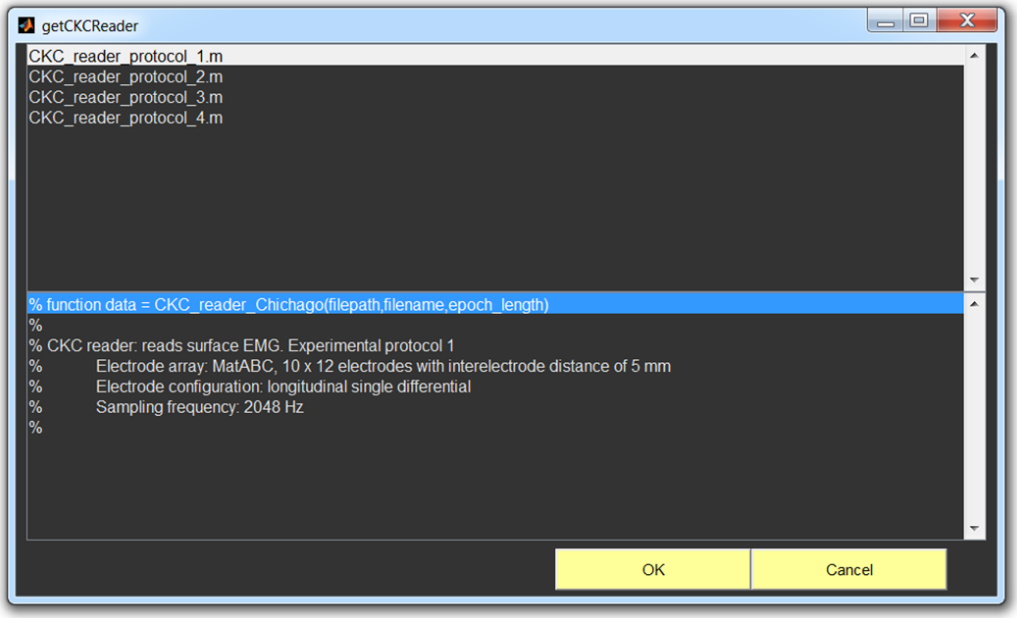


Figure 7: `getCKCReader` Dialog Window for selection of the reader of EMG signals. Upper panel displays all available readers. Lower panel displays

description of the currently selected reader. Selection of the reader is confirmed by pressing the `OK` button.

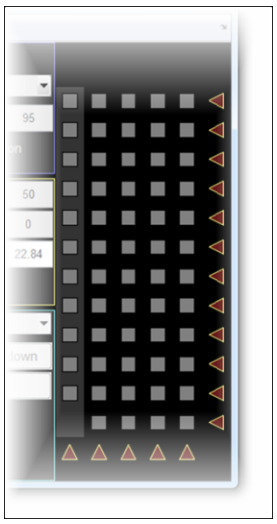


Figure 8: `Channel selection frame` displaying relative spatial organization of surface electrodes (grey rectangles). By clicking the red triangles, EMG channels corresponding to specific row or column of surface electrodes get selected for visual inspection. Selected column/row is denoted by dark grey rectangle.

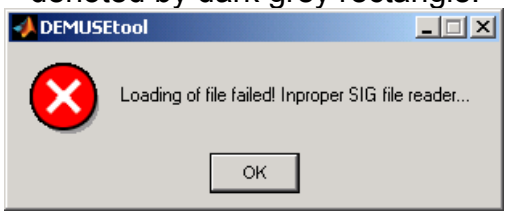


Figure 9: Error Dialog Window signalling the failure of EMG loading.

4.3 Filtering EMG signals

DEMUSE tool uses 2nd order Butterworth band-pass filters to filter the raw EMG signals. Filter's cut-off frequencies can be controlled by typing new values into the text labels shown in **Figure 10**. Default cut-off frequencies are set to 20 Hz and 500 Hz, respectively.

When the number of MUs in the EMG signals is high, it is beneficial to turn on the time differentiation of the EMG channels (**Figure 10**). Time differentiator is a high-pass filter which suppresses small MUAPs and enhances the discrimination of MUAPs from different MUs. Selection of time differentiator is optional and left to the user. A good practice is to play a bit with the time-differentiation and band-pass filtering before running the decomposition. The effect of time-differentiations and

band-pass filtering can be examined by plotting EMG channels and/or their power spectra (see Section 4.4).

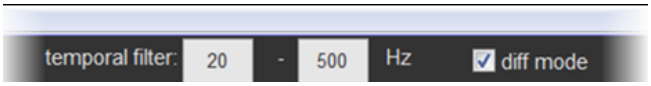


Figure 10: Text labels controlling the cut-off frequencies of built-in Butterworth band-pass filter (*left*) and check-box for selection of time differentiator (*right*). Time differentiator is a high-pass filter which suppresses the activity of small background MUAPs and enhances the differences between the MUAPs from different MUs.

Note: Both filters are implemented in `hpbutter.m` and `lpbutter.m` files and can be freely modified by more experienced users.

!

DEMUSE tool automatically removes line interference and tests the acquired EMG channels for presence of movement artefacts and bad skin-electrode contacts. However, percentage of the EMG channels to be included into the decomposition must be specified explicitly by changing the value of `auto sel. chs.` textbox (**Figure 11**). Setting the value to 95 %, for example, means that 5% of the channels with the lowest estimated signal quality will be discarded by the CKC decomposition.

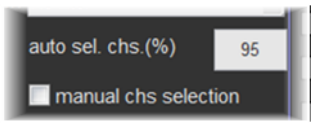


Figure 11: Text box for definition of EMG signal quality. DEMUSE tool automatically removes line interference and tests the acquired EMG channels for movement artefacts and bad skin-electrode contacts.

EMG channels can also be discarded manually. First select the `manual chs selection` checkbox (**Figure 11**). Second, simply click on grey rectangles representing the electrodes in `Channel selection frame` (**Figure 12**). Discarded channel is denoted by white rectangle crossed out by two red lines (**Figure 12**). To reselect the discarded EMG channel simply click again on the corresponding rectangle.

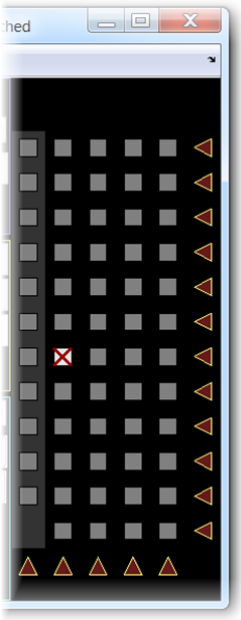


Figure 12: `Channel selection frame` displaying relative spatial organization of surface electrodes (grey rectangles). By clicking the corresponding gray rectangles, individual EMG channels can be discarded or selected again for decomposition. The discarded channel is denoted by white rectangle crossed out by two red lines.

Discarding of EMG channels influences the commands for plotting of EMG channels, estimated MUAPs and MUAP trains. For example, when '`plot MUAPs`' button is pressed, the MUAPs in the discarded EMG channels are not plotted.

4.4 Visualization of EMG signals

To display filtered EMG signals, first select the offset and the length of the signal interval (**Figure 13**, left bottom panel). Select the corresponding electrode row or electrode column by clicking on red triangle on the bottom or the left side of the `Channel selection frame` (**Figure 13**, left panel) and click on "`plot signals`" button (**Figure 13**, right panel). Matlab figure with selected EMG channels appears (**Figure 14**). Zoomed-in version of **Figure 14** is depicted in **Figure 15**.

Note: *Due to the large number of acquired EMG channels, only selected row/column of EMG channels can be displayed in one figure. The number of figures, however, is not limited. You can display all the EMG channels by consecutively selecting the different electrode columns, for example.*

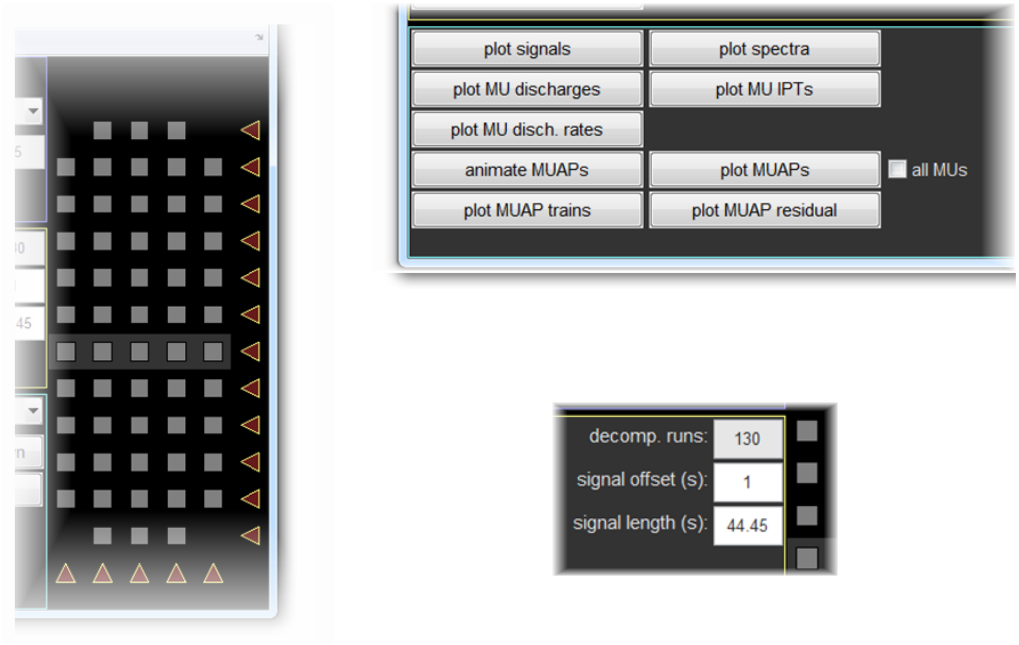


Figure 13: Channel selection frame with selected 7th row of electrodes (left panel), plot signals button (right upper panel) and editable text fields for selection of offset and length of displayed signal interval (right bottom panel).

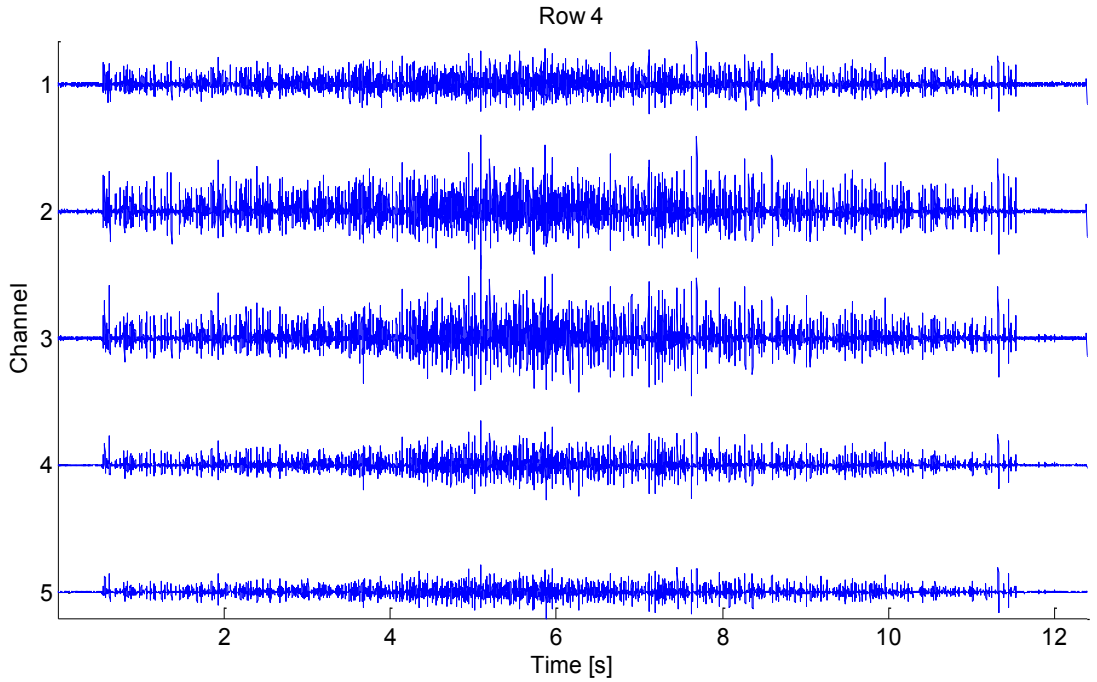


Figure 14: Matlab figure with selected EMG channels

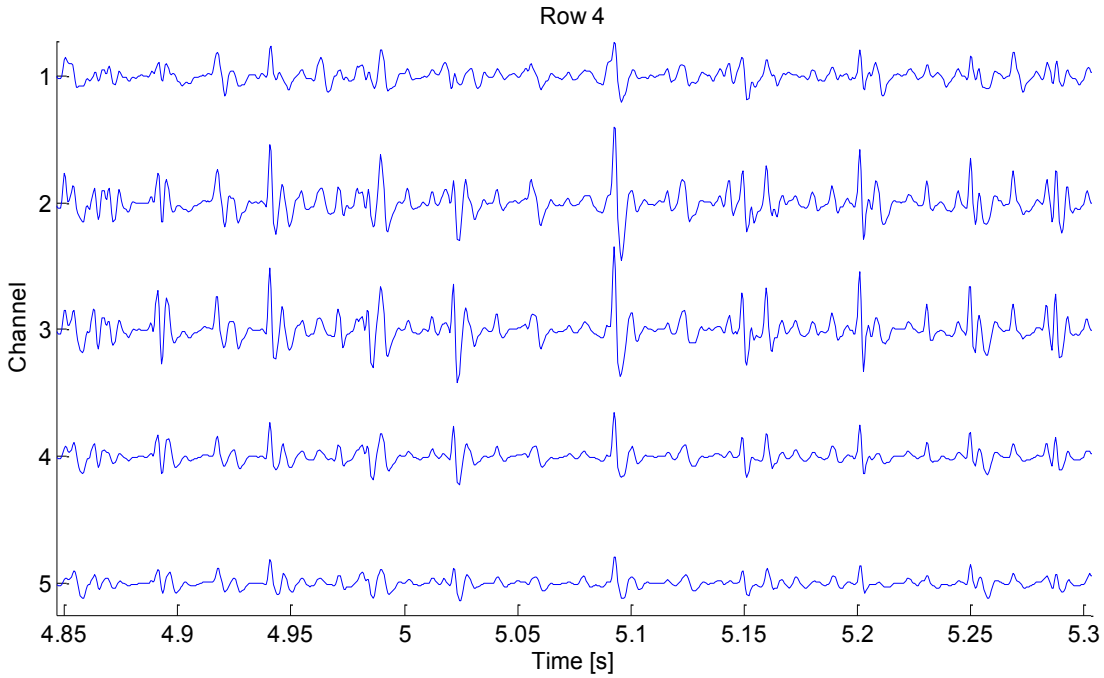


Figure 15: Zoomed-in version of **Figure 14**.

Displayed figures can be manipulated by using standard Matlab graphical tools for zooming in/out, for saving and printing the figure (**Figure 16**). Figures can be closed by clicking on a corresponding buttons in the top right corner of each figure (**Figure 17**).

Figure 16: Matlab figure toolbar with tools for zooming in/out on a figure, and for saving and printing the figure.

Figure 17: Buttons for minimization, maximization and closing of the figure.

To display power spectra of the EMG channels, press the "`plot spectra`" button (**Figure 13**, right panel). Matlab figure with spectra of selected EMG channels (as estimated by fast Fourier transform (fft) algorithm) appears (**Figure 18**).

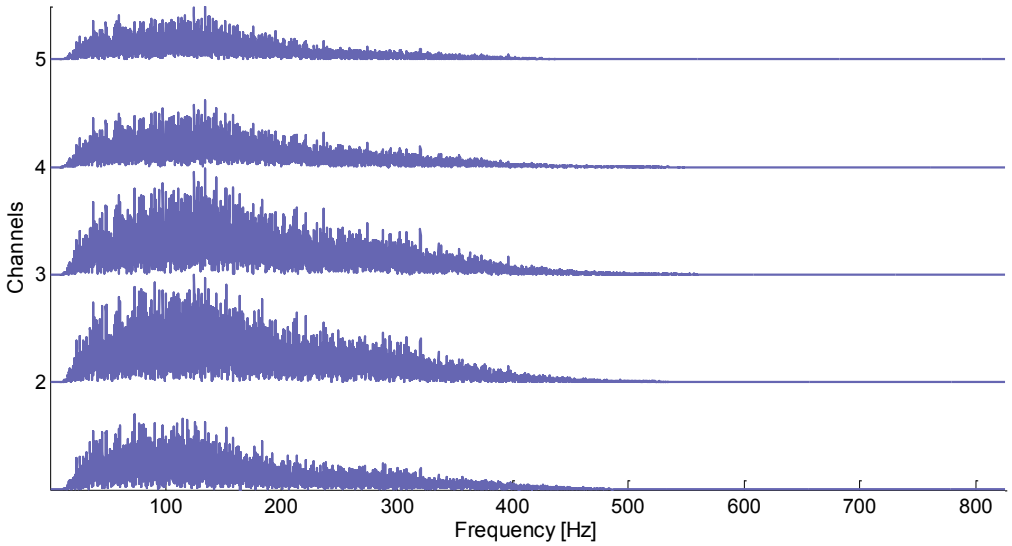


Figure 18: Matlab figure with power spectra of selected EMG channels

4.5 Decomposition of EMG signals

DEMUSE tool uses the gradient Convolution Kernel Compensation (gCKC) decomposition technique [3]. Decomposition is fully automatic and minimally biased by the properties of investigated muscle. The user specifies only the number of decomposition runs, the initial offset and the length of EMG time interval to be decomposed (**Figure 19**). The latter two parameters allow discarding the initial signal portions where, for example, the contraction level is not yet stabilized, etc. The initial signal offset and interval lengths are measured in seconds.

gCKC is a sequential MU identification technique and requires one iteration run per each identified MU. The user can predefine maximal number of iterations by changing the value in "`decomp. runs`" text field (**Figure 19**). As a general rule, the number of iterations should be larger or equal to the number of expected MUs (excluding the small and deep MUs, which are treated as the background noise). As the exact number of MUs is difficult to estimate, the number of decomposition runs should be large (default value is set to 30). DEMUSE tool automatically tracks the accuracy of MU identification (Section 4.6.6) and discards the results with low accuracy. As a result, the number of identified MUs is usually significantly smaller than the number of decomposition runs.

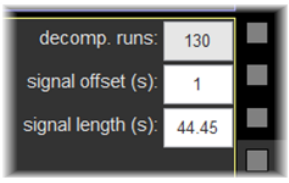


Figure 19: Editable text fields for selection of the number of decomposition runs, signal offset and signal length.

Note: *Due to the large number of EMG channels and high memory consumption of gradient CKC method, the length of decomposition interval should generally be limited to 20 s. Longer signals should be divided into 20s long epoch which should be decomposed independently. The optimal length of decomposition interval depends also on the amount the available computer memory.*

The decomposition starts by clicking on the "`decompose`" button and can be stopped by clicking the "`stop decomposing`" button (**Figure 20**).

Figure 20: Buttons for starting and stopping the decomposition.

When the decomposition ends, the decomposition results can be visualized and saved (Sections 4.8 and 4.9). Alternatively, one can change cut-off frequencies of band-pass filter or toggle time differentiation and rerun the decomposition by pressing the "`redecompose`" button (**Figure 20**). Contrary to the "`decompose`" button, "`redecompose`" button keeps the already identified MU discharge patterns and adds them to those reconstructed in the new decomposition run. On the other hand, the "`decompose`" button (**Figure 20**) automatically deletes previously identified MU discharge patterns and starts the decomposition from a scratch.

The EMG signals can also be decomposed in a batch mode, by clicking "`batch decompose`" button (**Figure 20**). The `Reader Dialog Window` (**Figure 7**) opens, offering the selection of CKC readers. After selecting the reader, the standard dialog box for selecting directory opens (**Figure 21**). Select the directory with the signals to decompose and click "`ok`" button. All the signals in the selected directory will be sequentially loaded into the DEMUSE tool and decomposed.

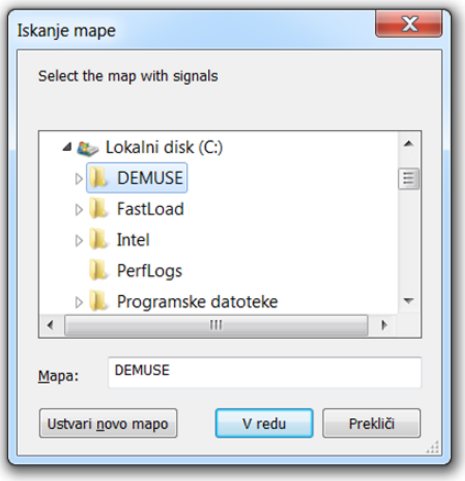


Figure 21: A dialog box for selecting directory with signals.

Note: *In batch decomposition mode, all the signals are decomposed using the same values of decomposition parameters (e.g., the number of decomposition runs, signal offset and length, filtering and time differentiation etc.). These values should be set before pressing the* "`batch decompose`" button.

The decomposition results are automatically saved in .mat files (within the selected directory with EMG signals). The name of the file with the results equals the name of the file with EMG signals with the suffix "`_offsetX_lengthY_runsZ.mat`" where X, Y and Z stand for the signal offset, length and number of decomposition runs. *Whenever the .mat file with that name already exists in selected directory with signals, the decomposition of the corresponding signal is skipped.*

Due to the large number of acquired surface EMG channels and high memory consumption of gradient CKC method, the length of decomposition interval should generally be limited up to ~60s (the exact value depends on the amount of available computer memory). Longer signals should be divided into 20-40s long epochs which should be decomposed independently. Alternatively, the MU signatures in the space of discharge patterns can be reconstructed on a portion of a signal (e.g., by running the decomposition on the first 20s long epoch) and then applied to the entire signal length (Section 4.6.4). This takes much less time than separate decompositions of different epochs, but is limited to MUs identified in the corresponding time epoch only. MUs recruited at latter time moments will not be identified. In other words, if we decompose first out of four consecutive force ramps in **Figure 22**, for example, and then click on "`track MUs`" button in CKC inspector (Section 4.6.4), we will quickly retrieve entire discharge patterns of MUs identified during the first ramp (**Figure 22**), but will fail to identify MUs that were recruited after the first ramp, i.e. in the second, third and/or fourth ramp only. A good practice is to decompose the signal epoch with the highest expected number of active motor units (i.e.,

the last ramp in **Figure 22**), and then click on "`track MUs`" button in CKC inspector.

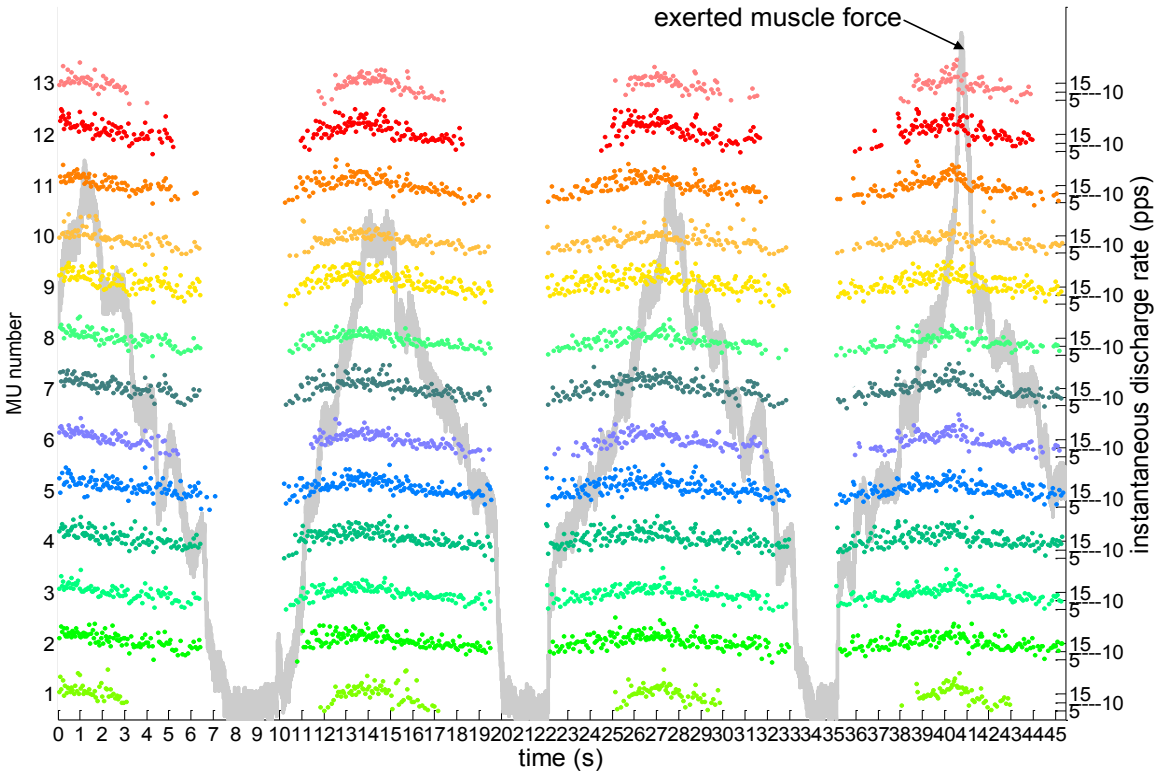


Figure 22: Result of "`Apply to entire signal`" button on an example of EMG signals recorded during four force ramps of abductor pollicis brevis muscle. Only the first force ramp was decomposed by clicking on "`decompose`" button, whereas discharge patterns on other three ramps were reconstructed by clicking on "`track MUs`" button. Each dot corresponds to a single MU discharge. Different MUs are denoted by different colours.

4.6 Editing and inspecting the decomposition results: CKC inspector

CKC inspector allows editing the raw outputs of the gradient CKC method, i.e., trains of delta pulses. Inspector is launched by clicking on the "`CKC inspector`" button (**Figure 23**).

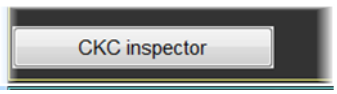


Figure 23: "`CKC inspector`" button.

CKC inspector window consist of three panels (**Figure 24**). The upper most panel is the MUAP panel. It displays multichannel MUAP of selected MU (as

detected by all surface electrodes). The central panel displays instantaneous discharge rate (IDR) and is called IDR panel. The reference signal (thin grey line in **Figure 24**) as determined by `ref_signal` in the CKC reader (Appendix I) is also displayed. The lower panel, so called innervation pulse train (IPT) panel, displays the train of MU discharge times as estimated by the gCKC decomposition technique (**Figure 25**).

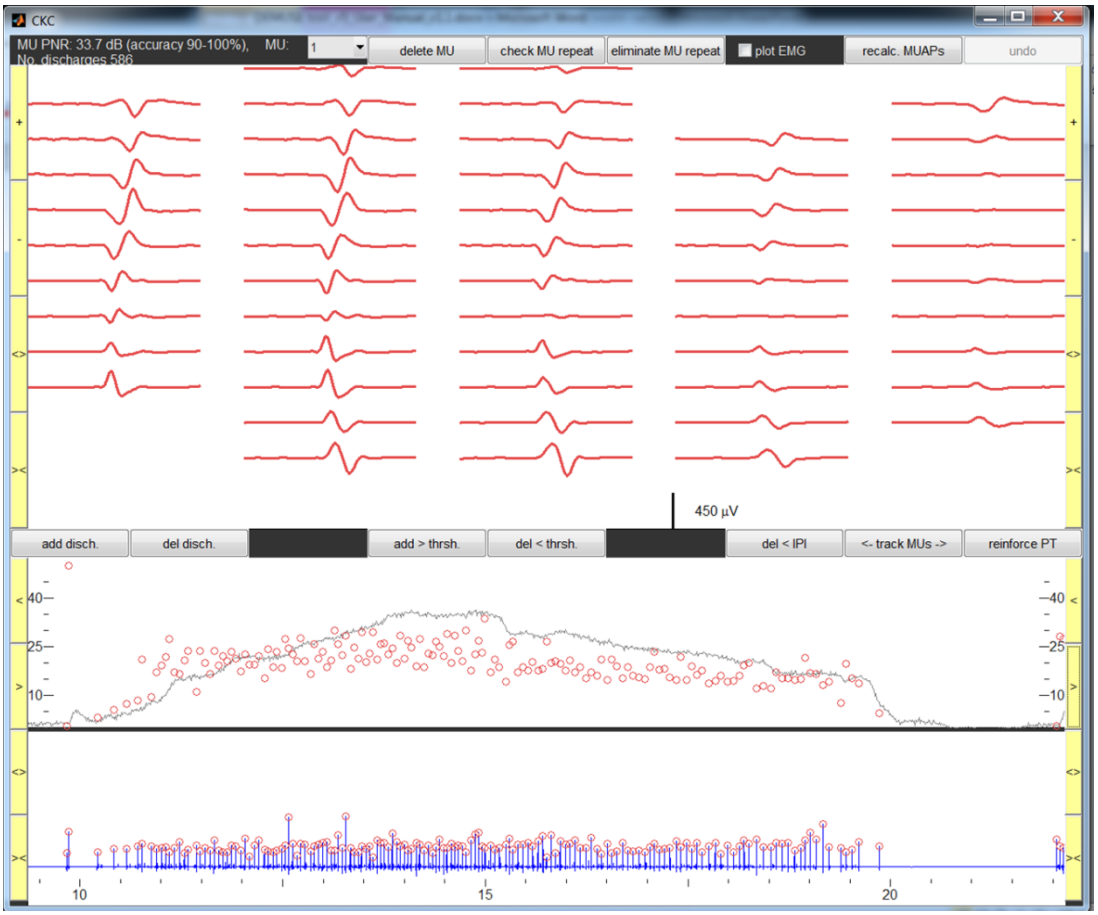


Figure 24: Window of "`CKC inspector`" with the MUAP panel (upper panel), IDR panel (central panel) and the lower IPT panel.

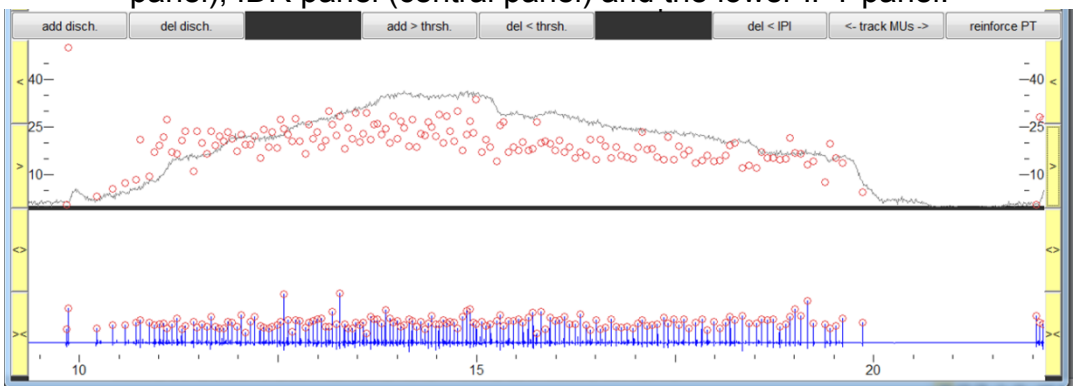


Figure 25: Lower panel of "`CKC inspector`" window with train of delta pulses as estimated by CKC method in the upper part of the panel, and instantaneous discharge rate plot in the lower part of the panel. In both plots,

horizontal ruler denotes the time (in seconds). Vertical axis of the lower plot denotes the instantaneous discharge rate (in pulses-per-second).

User can zoom-in and zoom-out on time axis by clicking on '><' and '<>' buttons, respectively (**Figure 26**). By clicking buttons '<' and '>' (**Figure 26**) the displayed portion of the MU discharge pattern is moved left and right, respectively. Zooming in and out on time axis can also be achieved by pressing `Ctrl-Z` and scrolling the mouse wheel. When combined with `Ctrl-M` key combination, scrolling the mouse wheel moves the time axis backward and forward, respectively.

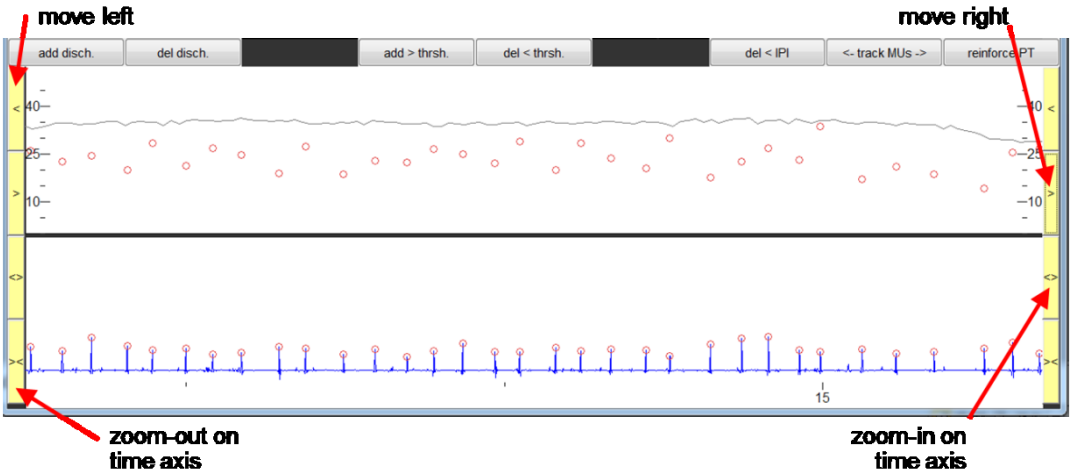


Figure 26: Same as in **Figure 25** with the time axis zoomed-in. Delta pulses denoting the discharge times of single MU are clearly visible. Base-line noise is negligible and inter-discharge interval exhibits regular behaviour. This gives us confidence in the results of CKC decomposition. Buttons on the edges of the lower panel control the size and position of the displayed portion of the MU discharge pattern.

4.6.1 Adding and deleting MU discharges

MU discharges can be added or deleted by clicking the '`add disch.`' and '`del disch.`' buttons. After each click on these buttons, mouse pointer changes from arrow to full cross. Drag the cross in the IPT panel to the pulse to be added/deleted and left-click to add/delete the MU discharge (**Figure 27**). Several MU discharges can be deleted simultaneously by clicking on '`delete < thrsh.`' button (**Figure 26**). In this case, full-cross pointer is used to determine the left and the right edge of the MU discharge cancellation interval in the IPT panel and the amplitude threshold of the pulses (vertical axis). All the discharges between the aforementioned edges and below the selected amplitude threshold will be deleted. Analogously, all the pulses above the selected amplitude threshold can be added by clicking

the ‘add > thrsh.’ button and selecting the right and the left edge of the MU discharge interval.

Figure 27: Portion of a MU discharge pattern before (*left panel*) and after cancellation of the central MU discharge (*right panel*). In the right panel, full-cross pointer used for selection of the MU discharge is partially visible. MU discharge is deleted by clicking on the ‘del disch.’ button, positioning the full-cross pointer over the MU discharge (i.e. pulse) and clicking the left mouse button.

The MU discharges can be deleted also in the IDR panel. To do so, click on the “del < IPI” button (**Figure 26**). The full-cross pointer appears. Select the right and the left edge of the MU discharge interval by two consequent left clicks in the IDR panel. The maximal vertical position of both clicks determines the maximal allowed instantaneous discharge rate (as displayed in the IDR panel). All the MU discharges between the selected right and left edge that exhibit the instantaneous discharge rate above the selected maximal one are deleted.

4.6.2 Estimation of MUAP templates

The MUAP panel of the "CKC inspector" window displays multichannel MUAP template of selected MU as detected by all the surface electrodes and estimated by spike triggered averaging of sEMG channels (red thick lines in **Figure 24**). When calculating the average MUAP on a given channel, all the MU discharges that are displayed in the IPT panel are used as triggers. Displayed MUAPs are spatially organized in rows and columns, reflecting the relative position the of pick-up electrodes.

Note: *The calculation of MUAP templates uses only the of MU discharges that are displayed in the IPT panel. Their number can be controlled by zooming in and out the time axis in the IPT panel. The larger the number of MU discharges used, more accurate the estimation of MUAP templates displayed in the MUAP panel.*

In order to speed-up the zooming in and out on the time axis in the IPT panel, the MUAP templates are recalculated only when 'recalc. MUAPs' *button is pressed (***Figure 24***).*

When plot EMG checkbox is checked (**Figure 28**), portions of the original surface EMG channels around that selected MU discharge are displayed aligned with the displayed MUAP templates in the MUAP panel (blue thin lines in **Figure 28**). The selected MU discharge is indicated by thick grey circle in the IPT panel (**Figure 28**). Any MU discharge can be selected by right-clicking on the corresponding red circle in the IPT panel (**Figure 28**).

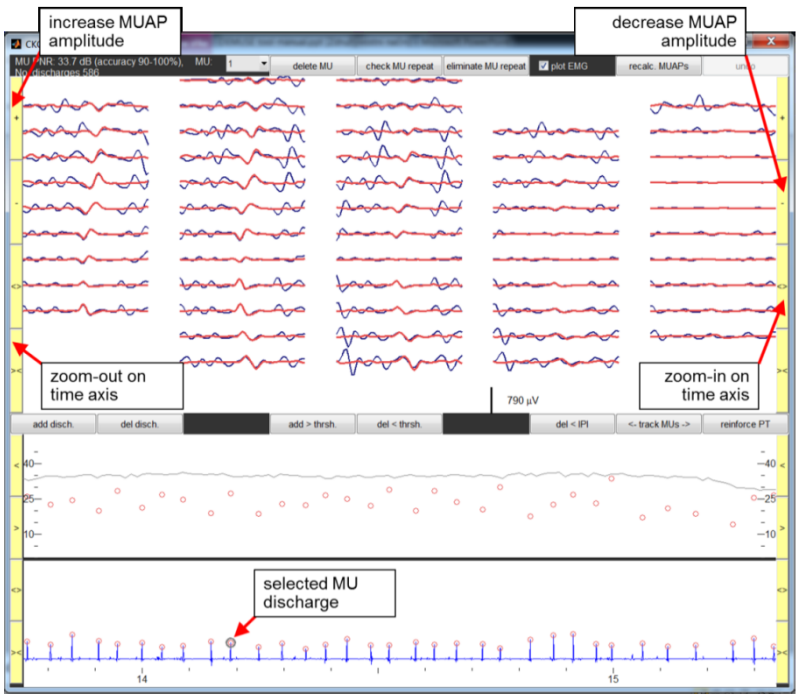


Figure 28: Buttons for controlling the length and the scale of the displayed multichannel MUAP template and raw sEMG portions. Buttons '+' and '–' scale the MUAP/EMG amplitude. Buttons '><' and '<>' zoom in and out on the time axis, respectively.

The '+', '−', '><' and '<>' buttons in the MUAP panel control the length and the scale of displayed MUAP templates. Their use is further explained in **Figure 28**.

4.6.3 Improvement of decomposition results

DEMUSE tool uses the gradient Convolution Kernel Compensation technique (gCKC) [Holobar et al. 2007a,b, 2012, 2014] to iteratively identify the discharge patterns of each individual MU. Although theoretically proven to result in Bayesian optimal estimate of MU discharge patterns, the gCKC occasionally requires relatively large number of iterations to converge to the final solution. In order to speed up the decomposition process, the maximal number of gCKC iterations is currently limited to 45 per MU. This suffices for accurate identification of majority of MUs as gCKC typically converges in about 30 iterations. However, further improvements of the identified MU discharge pattern are possible by clicking the '`reinforce PT`' button in CKC inspector. Each click on the button executes one gCKC iteration (**Figure 29**).

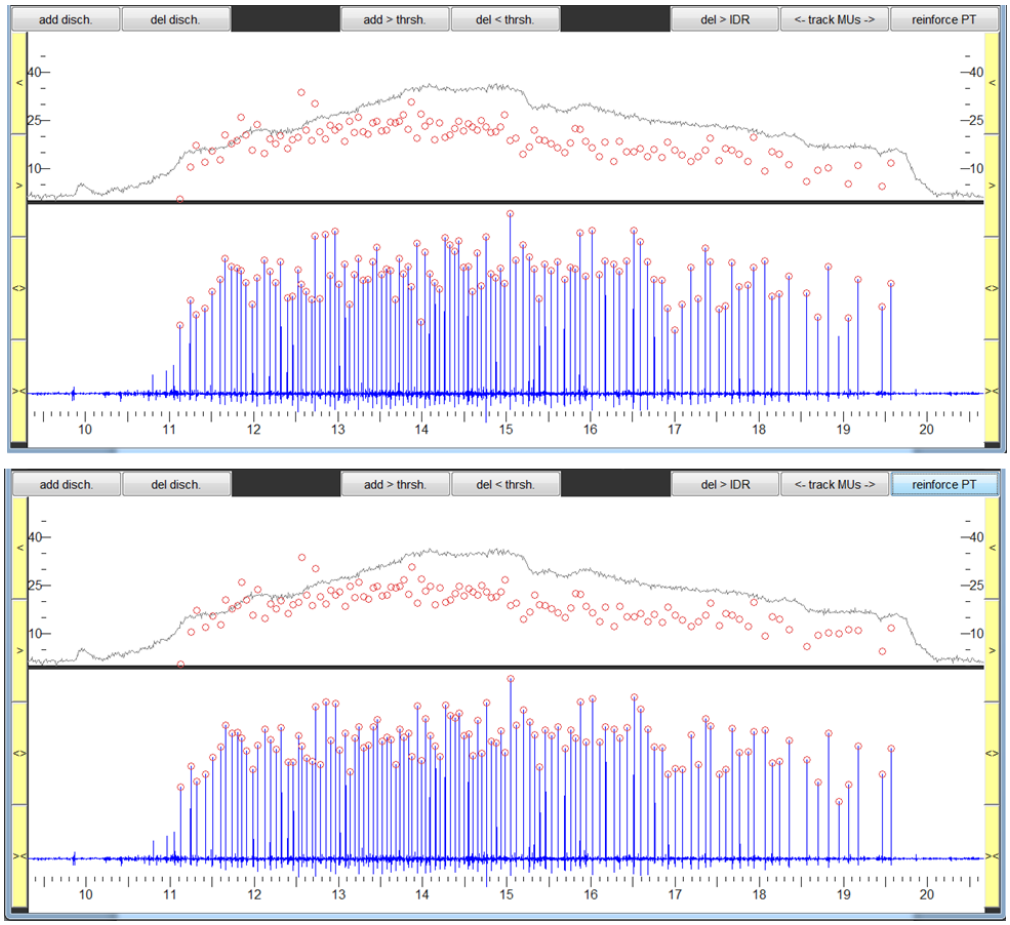


Figure 29: MU discharge pattern before the '`reinforce PT`' button is pressed (upper panel) and afterwards (lower panel). Each click on the '`reinforce PT`' button executes on iteration of gCKC technique.

4.6.4 MU tracking

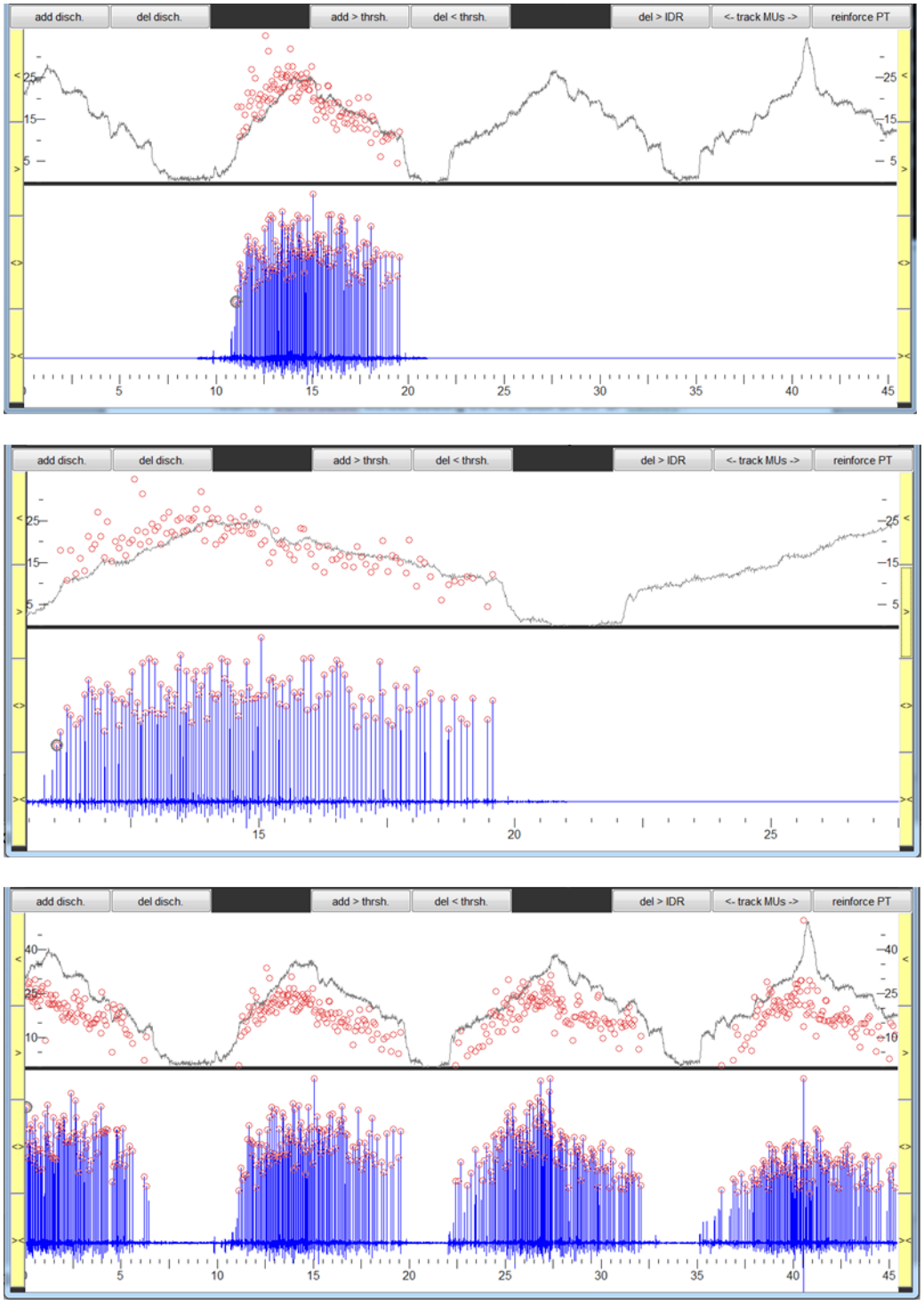


Figure 30: MU discharge pattern as identified by gCKC technique on time interval from 10th to 20th s (*upper panel*); The selection of time interval for MU tracking (*central panel*) - as a rule of thumb, the selected interval should ~ 20s long; The MU discharge pattern after MU tracking (*lower panel*).

The MUs identified on a portion of EMG signal (e.g., by running the decomposition on the first 20 s long epoch of 100 s long EMG signal) can be automatically tracked on the remaining parts of the signal. First, zoom-in/move-to the time interval with the identified MU discharge patterns in the IPT panel (**Figure 30**, central panel). Click on "`track MUs`" button to iteratively identify the MU discharges on the entire time support of EMG signal. In each iteration step, the CKC inspector searches for the discharges of all the identified MUs on the displayed time interval and automatically moves forward/backward from the current position for half of the length of displayed time interval until the end/beginning of the EMG signal is reached and the entire discharge patterns of all the MUs are identified. This procedure can take several minutes.

Notes: *MU tracking is currently applied to all motor units, i.e. one click on* "`track MUs`" button causes the tracking of all MUs.

! *CKC inspector uses discharges of individual MU in the selected time interval to construct the MU specific filter and track the MU.* The quality of MU filter is directly proportional to the number of MU discharges on the selected interval. *Thus, all the MUs should be active in the selected time interval before the* "`track MUs`" button is clicked*. As a rule of thumb, the selected time interval should be at least 20 s long and the MUs should be active for at least 10 s.* MUs that are not active on the selected time interval will not be tracked. MUs that are initially active for less than 10 s will be tracked with lower accuracy.

MU tracking with selected intervals of length 40 s or more requires large amount of computer memory and can result in 'out of memory' exception. This depends on the amount of available memory.

4.6.5 Managing the identified MUs

The upper row of CKC inspector window contains dropbox for selection of the currently displayed MU and the following buttons: '`delete MU`', '`check MU repeat.`', '`eliminate MU repeat.`', '`recalc. MUAPs`' and '`undo`'. (**Figure 31**) The '`recalc. MUAPs`' button recalculates the MUAPs of currently selected MU and was already described in Section 4.6.2. The '`delete MU`' button deletes the currently selected MU.

The '`check MU repeat.`' button compares the discharge pattern of currently selected MU with discharge patterns of all other MUs and reports the list of MUs that share at least 10 % of discharges with currently selected MU. This list is empty when there is no such MU.

MU that share too many discharges with the currently selected MU can be deleted manually (i.e., by pressing the '`delete MU`' button). Alternatively,

they can be removed automatically by pressing the 'eliminate MU repeat.' Button. In such cases, the CKC inspector automatically identifies and groups all the MUs that share more than 30% of discharges. Only one MU per group (the one with the highest PNR metric) is maintained, whereas all the other MUs are discarded.

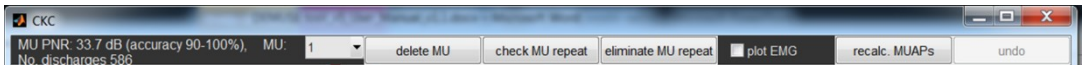


Figure 31: the upper row of CKC inspector with controls for selection and management of MUs.

Note: During the decomposition process, the DEMUSEtool automatically identifies and groups all the MUs that share more than 30% of discharges. Only one MU per group (the one with the highest PNR metric) is maintained, whereas all the other MUs are discarded. However, manual editing of decomposition results and further calls of gCKC method (e.g., by pressing either the 'reinforce PT' or the "track MUs" button) can lead to identification of already identified MU. Regular use of the 'check MU repeat.' and , 'eliminate MU repeat.' buttons is, thus, recommended.

4.6.6 Assessing the accuracy of decomposition

DEMUSE tool and CKC inspector calculates the Pulse-to-Noise Ratio (PNR) metric for each identified motor unit and, thus, enable automatic assessment of accuracy in motor unit identification. Contrary to the other state-of-the art metrics in the field of EMG decomposition, the PNR metrics does not rely on regularity of motor unit discharge pattern. This makes it plausible candidate for assessment of quality of surface EMG decomposition in the case of various pathologies, such as pathological tremor, where the regularity of motor unit discharge patterns cannot be guaranteed. As verified on extensive simulation studies [Holobar et al. 2014], motor units with PNR > 30 dB exhibit sensitivity in identification of MU discharges > 90% and false alarm rates < 5 % (**Figure 32**).

Note: For the PNR metric to accurately reflect the sensitivity and false alarm rates, the number of MU discharges must be greater than 30. The larger the number of MU discharges, more accurate is the PNR metric.

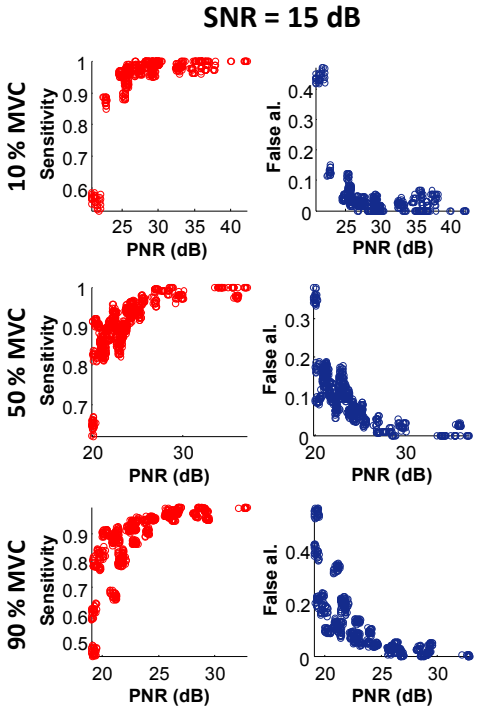


Figure 32: Sensitivity and false alarm rate in identification of motor unit discharges from surface EMG as a function of pulse-to-noise ratio (PNR) metric and muscle excitation level (MVC). Results were averaged across 20 realizations of synthetic surface EMG signals.

4.6.7 Closing the CKC inspector window

CKC inspector maintains its local copy of decomposition results. This local copy can then be transferred back to the DEMUSEtool by simply closing the CKC inspector window. The following dialog window appears

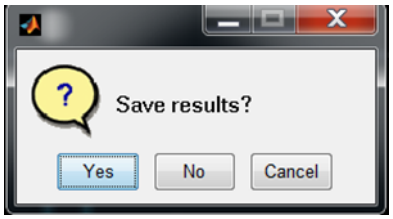


Figure 33: Dialog window appears whenever the CKC inspector is closed.

Press '`Yes`' button to save the edited decomposition results to DEMUSE tool and close the CKC inspector. Press '`No`' button to close the CKC inspector without saving the results back to DEMUSE tool. Press '`Cancel`' button to cancel the operation (CKC inspector remains open).

When saving the results from CKC inspector (i.e., when '`Yes`' button is pressed in **Figure 33**), the DEMUSE tool automatically identifies and groups

all the MUs that share more than 30% of discharges. Only one MU per group (the one with the highest PNR metric) is maintained, whereas all the other MUs are automatically discarded. This prevents the multiple identifications of the same MU.

4.7 Deleting the identified MUs

Identified discharge patterns of specific MU can be deleted by first selecting the MU in a "`selected MU`" pup-up menu and then clicking on a "`delete MU`" button (**Figure 34**). A window opens for conformation of MU cancellation (**Figure 35**). To delete the MU, click on '`Yes`' button. To return to DEMUSE tool without deleting the MU, click on '`No`' or '`Cancel`'.

Figure 34: "`delete MU`" button and "`selected MU`" pup-up menu.

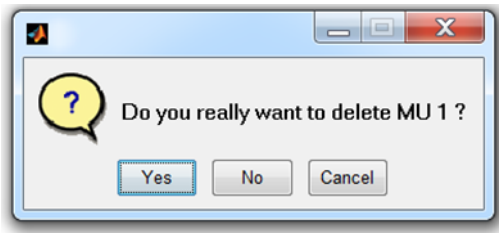


Figure 35: window for conformation of MU deletion.

4.8 Plotting the decomposition results

The user can plot the discharge patterns of identified MUs, instantaneous and smoothed MU discharge rates, multichannel MUAPs and reconstructed MUAP trains. In addition, MUAP generation, propagation and attenuation can be animated for each identified MU. All the graphical results are depicted in Matlab figures and can be easily manipulated by standard Matlab's editing tools. Background colour of all the plots can be selected in "`Properties`" menu (**Figure 36**). The size of fonts in figures can be set by changing the '`Figure Font Size`' property (**Figure 36**).

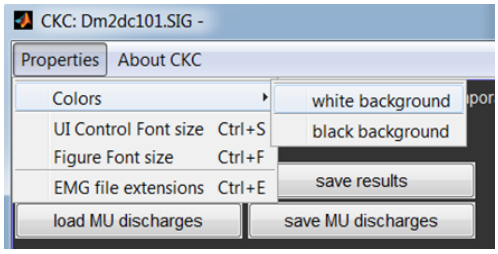


Figure 36: "`Properties`" menu allows selection of a background colour of all the plots, of CKC inspector and of sEMG editor.

4.8.1 MU discharge patterns plots

MU discharge patterns are plotted by pressing the "`plot MU discharges`" button (**Figure 37**). A Matlab figure opens with a plot of all reconstructed MU discharge patterns (**Figure 39**). Each circle in the figure corresponds to a single MU discharge. The horizontal position of the circle denotes the time of MU discharge, whereas its vertical position reflects instantaneous MU discharge rate (calculated as a quotient between the sampling frequency and the inter-pulse interval preceding the given MU discharge). Discharge patterns of different MU are depicted one above the other. The reference signal (grey line in **Figure 39**) as determined by `ref_signal` in the CKC reader (Appendix I) is also displayed.

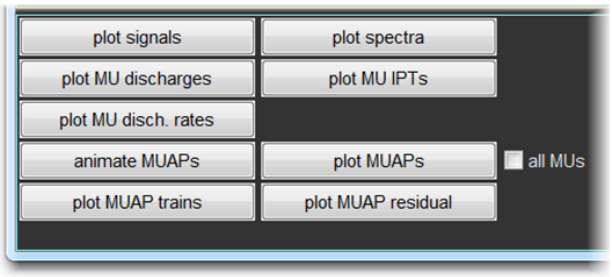


Figure 37: "`Plot MU discharge`" button

Before plotting, MUs can be sorted with respect to the number of their discharges (**Figure 37**, left panel) or with respect to their identification accuracy as assessed by PNR metric (**Figure 37**, right panel). Currently selected MU can be moved upwards and downwards by pressing the `MU up` and `MU down` buttons.

Figure 38: Sorting of identified MUs

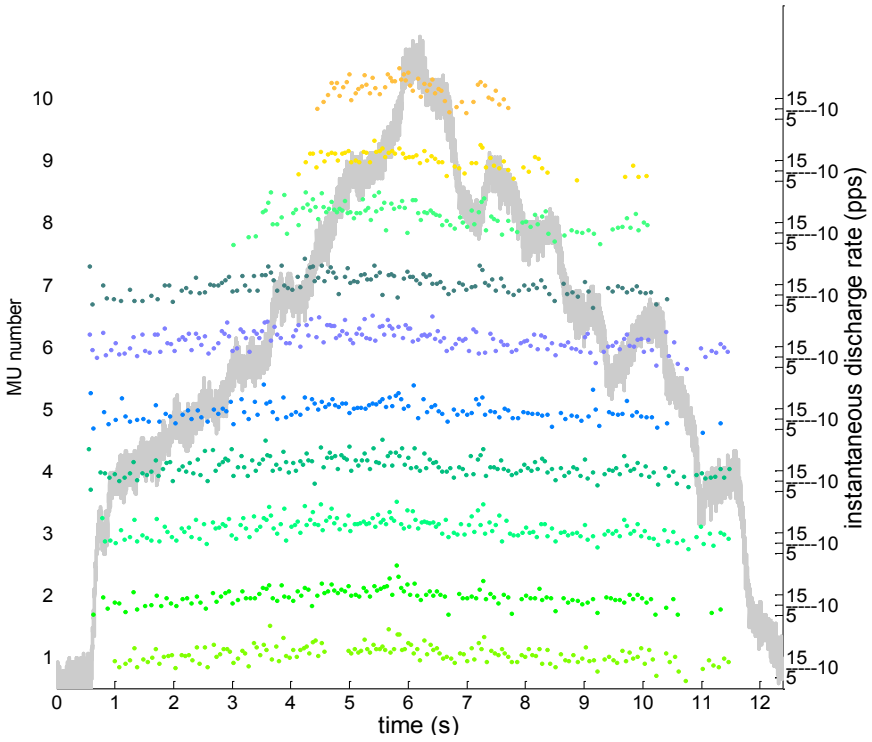


Figure 39: Plots of reconstructed MU discharge patterns. Vertical axis on the left displays MU IDs, vertical axis on the right displays the instantaneous discharge rates (in pulses per second - pps). The tick lines on the right denote the discharge rates of 5, 10 and 15 pps, respectively. Discharge patterns of different MUs are depicted one above the other.

4.8.2 Smoothed discharge rate plots

Smoothed MU discharge rates are plotted by pressing the "`plot disch. rates`" button (**Figure 37**). A Matlab figure opens (**Figure 40**) with a different colour lines depicting the smoothed discharge rates of different MUs (one line per each MU). The thick grey line depicts the reference signal as determined by `ref_signal` variable in the CKC reader (Appendix I). Smoothed discharge rates are calculated by low-pass filtering of the instananeous discharge rates (1st order Butterworth filter with cut-off frequency set to 2 Hz).

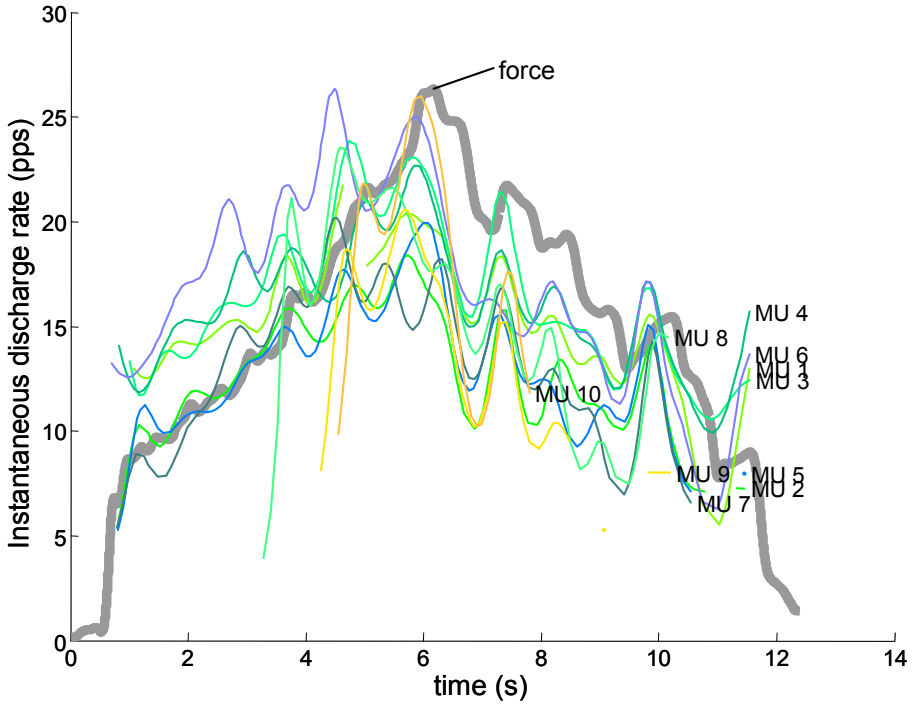


Figure 40: Plot of instantaneous discharge rates (coloured thin lines). Vertical axis depicts the instantaneous discharge rates (in pulses per second - pps). Thick grey line depicts the reference signal as determined by `ref_signal` variable in the CKC reader.

4.8.3 Multichannel MUAP plots

Multichannel MUAP plots (so called MU fingerprints) can be plotted by clicking on a "`plot MUAPs`" button (**Figure 37**). MU to be depicted is selected in a "`selected MU`" pup-up menu (**Figure 41**). A Matlab figure opens (**Figure 42**) with MUAP shapes as estimated by a spike triggered averaging of each acquired surface EMG channel. Displayed MUAPs are spatially organized in rows and columns, reflecting the relative position of pick-up electrodes.

Figure 41: "`selected MU`" pup-up menu.

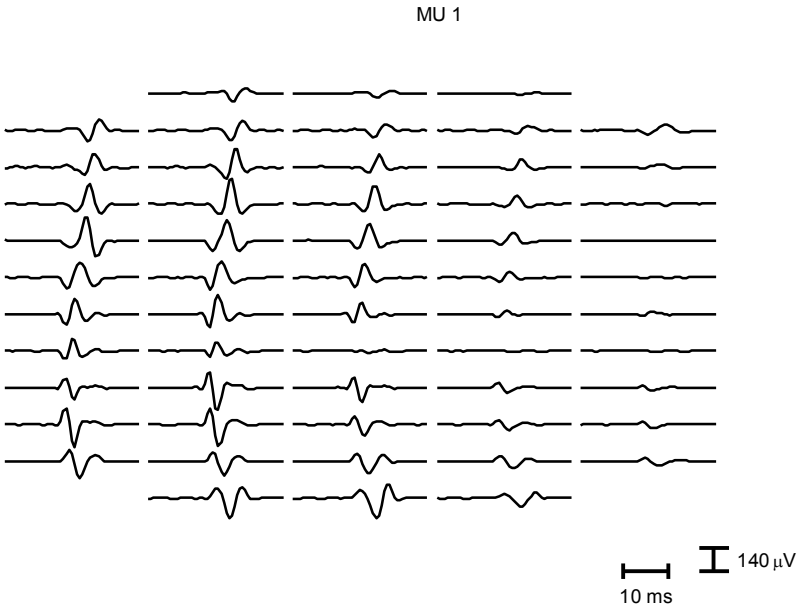


Figure 42: Multichannel MUAPs of MU 4 estimated by spike-triggered averaging of sEMG signals. sEMG signals were recorded with a grid of 61 electrodes arranged in 5 columns and 13 rows. The location of the innervation zone, tendon regions and propagation of motor unit action potentials are visible.

When '`all MUs`' checkbox is checked (**Figure 37**), the "`plot MUAPs`" button plots multichannel MUAPs of all the identified MUs (**Figure 43**).

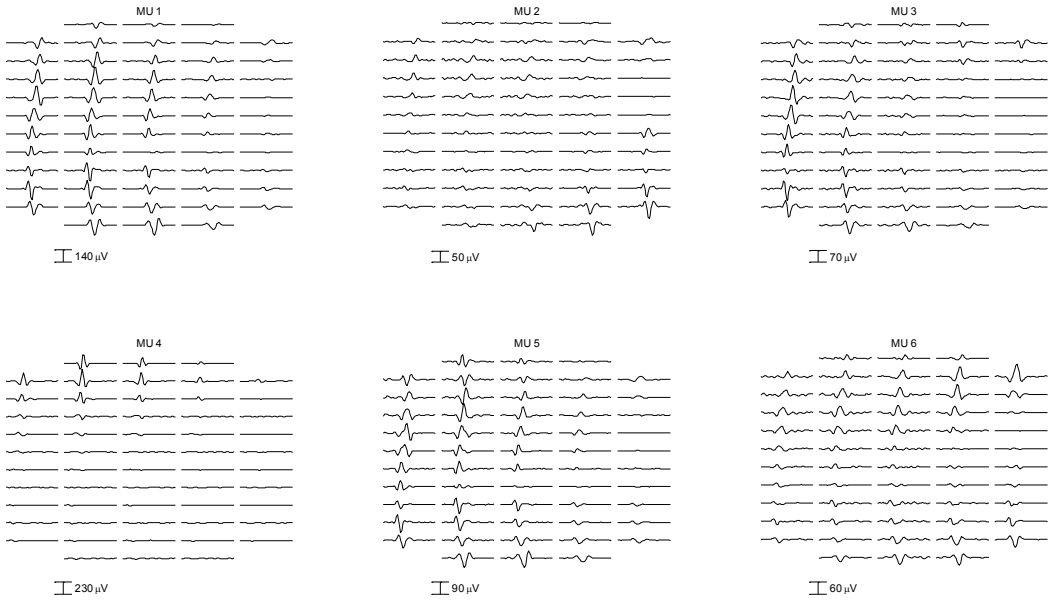


Figure 43: Multichannel MUAPs of estimated by spike-triggered averaging of EMG signals. sEMG signals were recorded with a grid of 61 electrodes arranged in 5 columns and 13 rows. The location of the innervation zone, tendon regions and propagation of motor unit action potentials are visible.

4.8.4 3D MUAP map animation

DEMUSE tool supports 3D animation of MUAP generation, propagation and attenuation. MUAP templates are first estimated by a spike triggered averaging of surface EMG channels. The channels are spatially organized into a discrete 2D map, reflecting the relative position of pick-up electrodes. The amplitudes of MUAP templates at a given time instant specify the height on this 2D map of channels. Missing intermediate points on the map are calculated by bilinear interpolation of MUAP amplitudes in four adjacent surface EMG channels. In the next animation frame, the time is moved forward by one signal sample and the 3D map is recalculated.

To start the 3D animation, select the MU (**Figure 41**). After clicking on the "`Animate MUAP`" button (**Figure 37**) the animation window opens (**Figure 44**) and the animation automatically starts. The animation begins approx. 5 ms before the actual generation of the MUAP and ends approx. 5 ms after the MUAP attenuation.

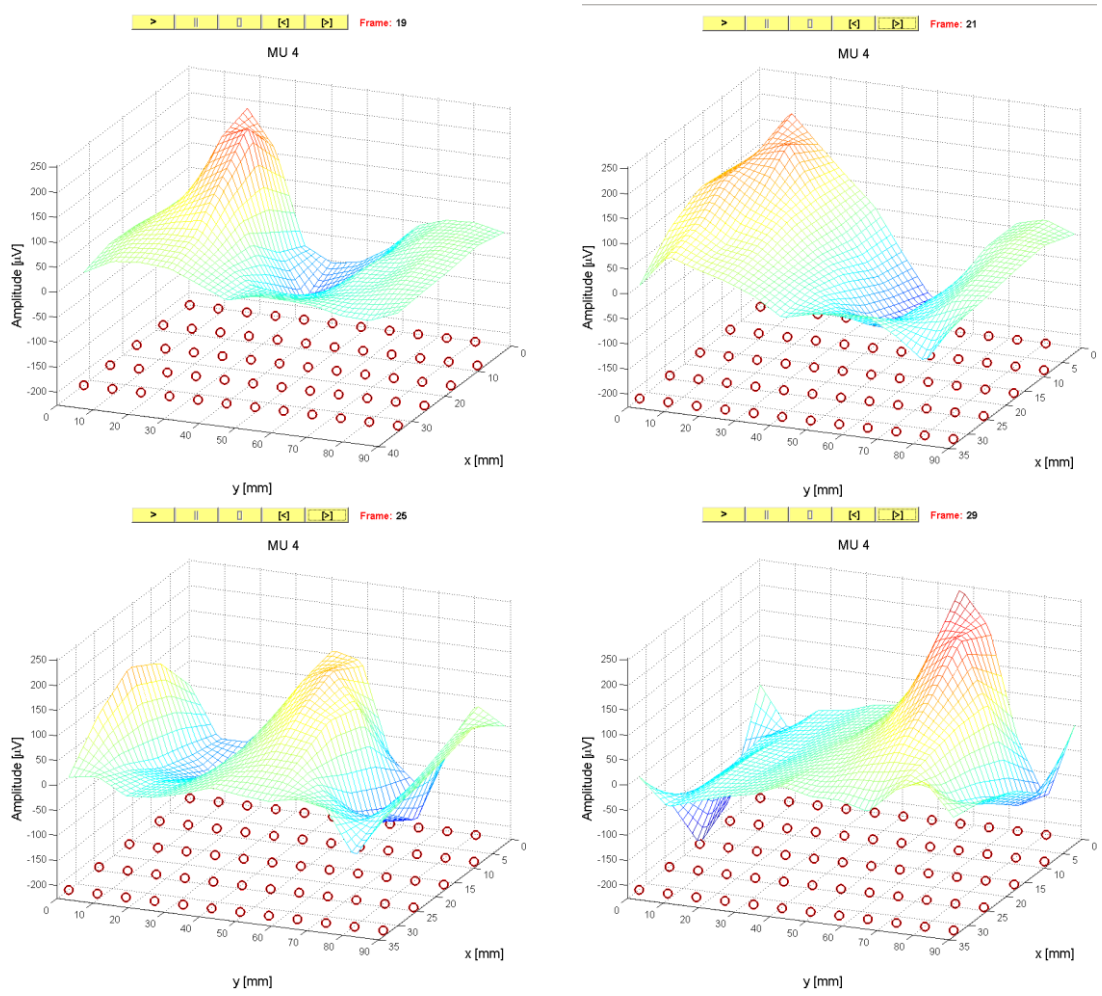


Figure 44: 3D animation of MUAP generation, propagation and attenuation. MUAP amplitudes on different EMG channels (red circles) specify the height of corresponding points on a 2D map (heights of intermediate map points are calculated by the bilinear interpolation of the MUAP amplitudes in four adjacent EMG channels).

Buttons on the top of the animation window (**Figure 45**) enable the following actions:

- “>” button: (re-)plays the animation;
- “| |” button: pauses the animation;
- ”[ ]” button: stops the animation;
- “[<]” button: animates the previous animation frame (i.e., step backward);
- “[>]” animates the next animation frame (i.e., step forward).

During the 3D animation, the axes of the 3D plot can be freely rotated. To rotate a 3-D axes, click on the axes and drag the cursor in the direction you want to rotate. When you release the mouse button, DEMUSE tool redraws the axes in the new orientation (**Figure 45**).

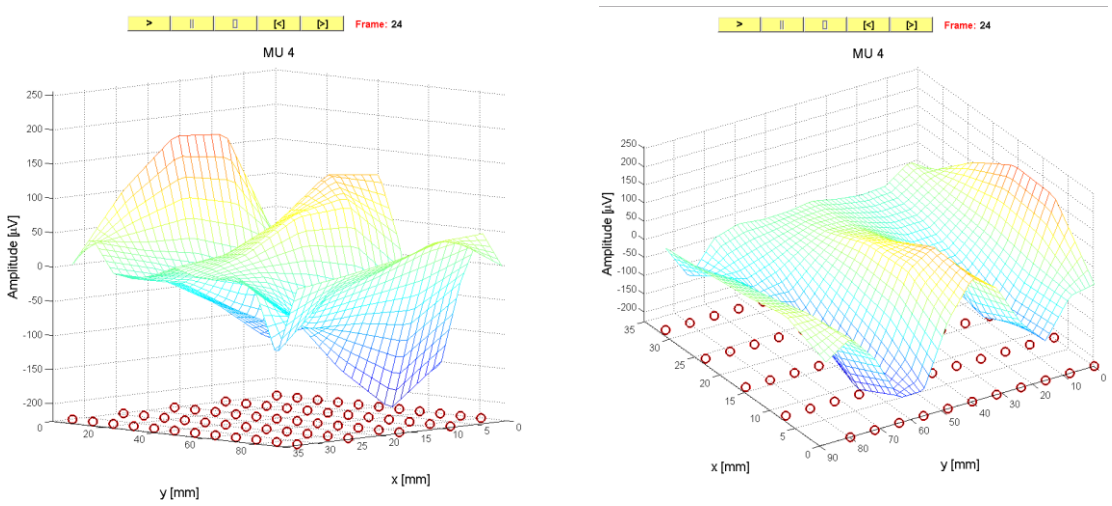


Figure 45: 3D rotation of the axes; the axes of the 3D plot can be rotated by dragging the cursor.

4.8.5 Plots of reconstructed MUAP trains

DEMUSEtool provides tool for plotting the sum of reconstructed MUAP trains superimposed to the original surface EMG signals. This proves beneficial when evaluating the efficiency of the decomposition process. In surface EMG, there are many small and deep MUs which cannot be identified. They contribute the background (physiological) noise. The second source of noise is so called instrumentation or thermal noise, which originates from the instrumentation’s parasite capacities, line interference, etc. All together, these sources add to the measurement noise and affect the efficiency of the EMG decomposition. By comparing the sum of the

reconstructed MUAP trains to the original EMG signal one can estimate the proportion of identified EMG energy.

In the DEMUSE tool, the reconstructed MUAP trains are calculated as follow. Firstly, the MUAP shapes are estimated by spike triggered averaging of the acquired surface EMG channel, using the identified MU discharge instants as triggers. The estimated MUAP shapes are then convolved with the identified MU discharge patterns and summed together. The sum of MUAP trains is subtracted from the original EMG signals and the following signal-to-interference ratio (SIR) between the original EMG signal and the residue after the subtraction is calculated:

$$SIR(i) = \left(1 - \frac{E\left[\left(x_i(n) - \sum_j z_{ij}(n)\right)^2\right]}{E\left[x_i^2(n)\right]}\right) \cdot 100\,\%$$

where $x_i(n)$ denotes the *i*-th EMG channel, $z_{ij}(n)$ stands for the *j*-th MU's MUAP train reconstructed from the *i*-th EMG channel and E stands for sample mean. Finally, the range of SIRs of EMG channels is displayed together with the reconstructed MUAP trains (**Figure 47**).

To display reconstructed MUAP trains, select the corresponding electrode row or electrode column (**Figure 46**, *left panel*) and click on "`Plot MUAP trains`" button (**Figure 46**, *right panel*). Matlab figure with selected sEMG channels and corresponding MUAP trains appears (**Figure 47**).

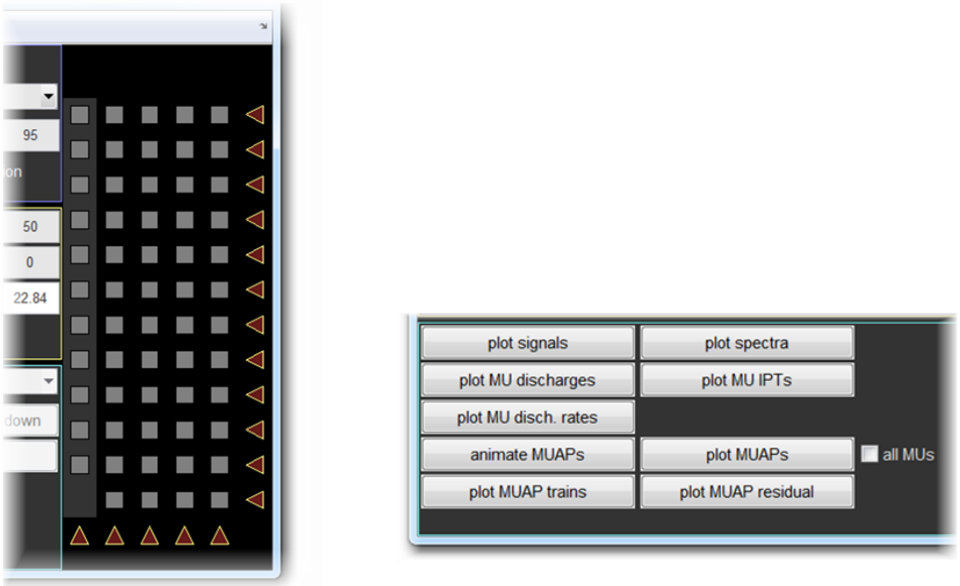


Figure 46: Channels selection frame (*left*), "`plot MUAP trains`" button and "`plot MUAP residual`" button (*right*).

Note: *Due to the large number of acquired EMG channels, only selected row/column of EMG channels can be displayed in one figure. The number of figures, however, is not limited. You can display the MUAP trains on all the EMG channels by consecutively selecting the different electrode columns, for example.*

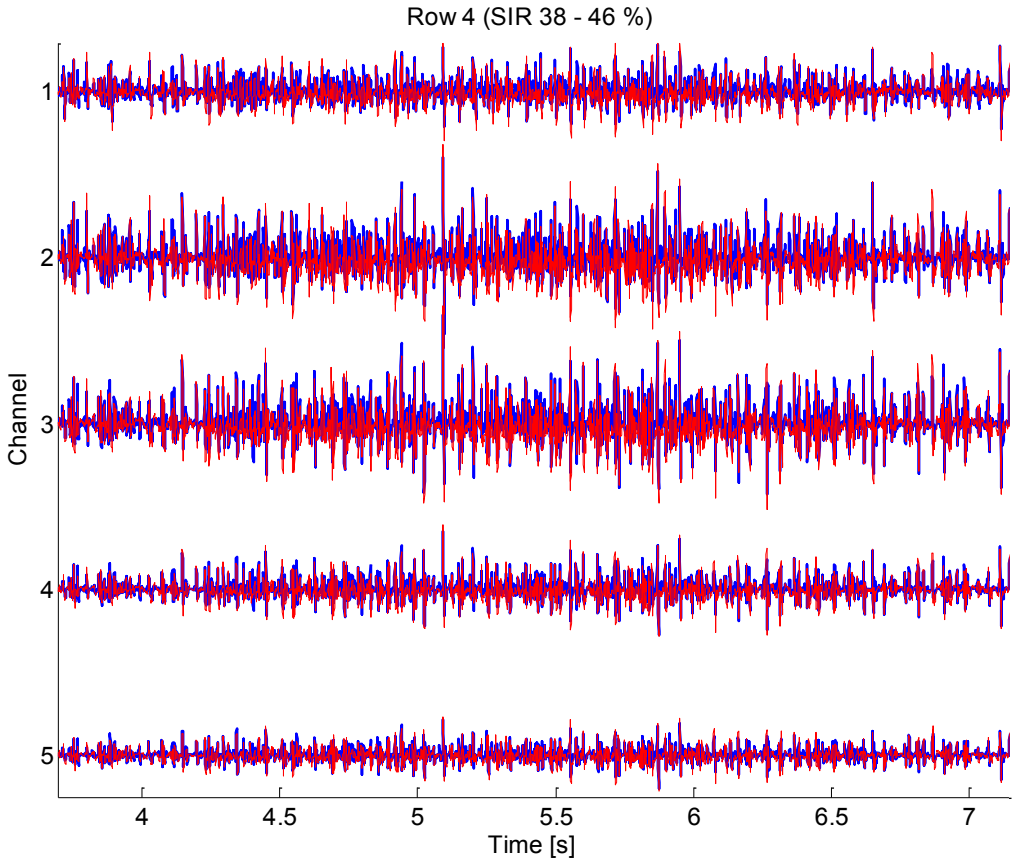


Figure 47: Matlab figure of selected surface EMG channels (*blue lines*) and corresponding reconstructed MUAP trains (*red lines*). The range of SIRs of the depicted channels is displayed on the top of the figure.

Plots of reconstructed MUAP trains are displayed as Matlab figures and can be freely manipulated by Matlab figure editing tools (i.e., figure resizing, zooming, rotating, printing, etc.). Zoomed-in portion of **Figure 47** is depicted in **Figure 48**. The user is referred to Matlab's documentation for further details on the use of the Matlab's graphic user interface.

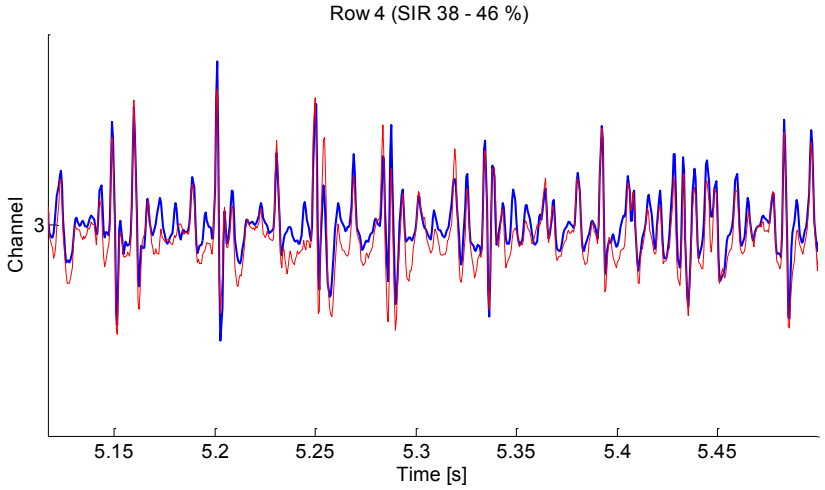


Figure 48: Matlab figure with selected surface EMG channels and corresponding MUAP trains (short signal segment from the signal shown in **Figure 47**)

Click on "`Plot MUAP residual`" button (**Figure 46**, *left panel*) opens the Matlab figure with selected surface EMG channels and corresponding residuals after subtraction of estimated MUAP trains (**Figure 49**). Range of SIRs of displayed EMG channels is displayed at the top of the figure.

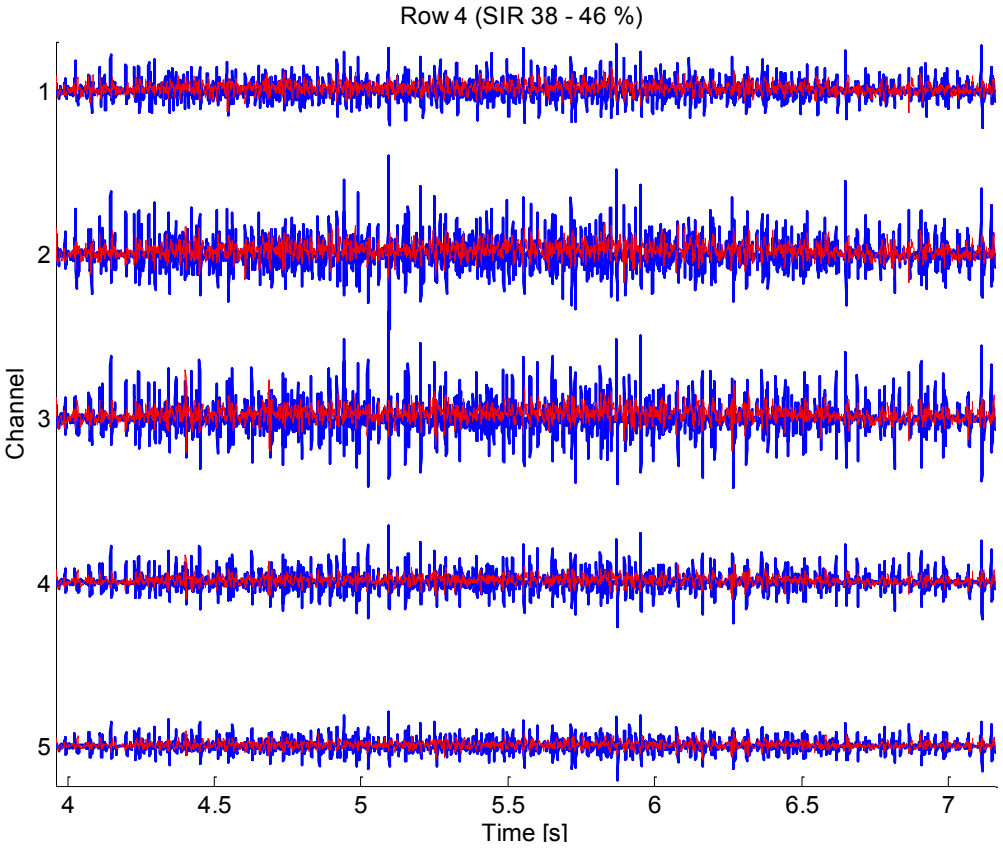
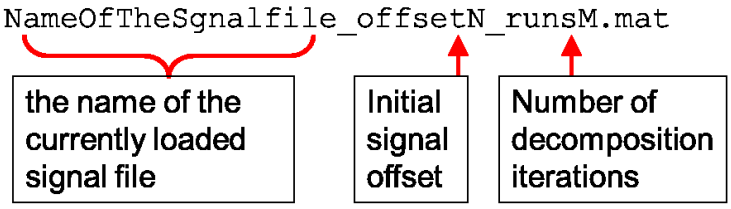


Figure 49: Matlab figure of selected surface EMG channels (*blue lines*) and residual after subtraction of reconstructed MUAP trains (*red lines*). The range of SIRs of the depicted channels is displayed on the top of the figure.

4.9 Saving and reloading the decomposition results

Decomposition results can be saved by clicking on the "Save results" button (**Figure 50**). The results are automatically saved into the directory containing the currently loaded signal file. The following file naming convention is used:


NameOfTheSgnalfile_offsetN_runsM.mat
the name of the currently loaded signal file
Initial signal offset
Number of decomposition iterations

where NameOfTheSignalfile stands for the name of the currently loaded signal file, N is the initial signal offset (in seconds) and M is the number of decomposition runs (see Section 4.5 for details). For example, the decomposition results of a signal file Subject1.SIG with initial signal offset set equal to 0 and number of decomposition iterations set equal to 30 is saved in the following Matlab file:

Subject1_offset0_runs30.mat

Saved results can be reloaded by clicking on the "load results" button (**Figure 50**). "Load results" dialog window opens (**Figure 51**). Choose the *.mat file and click on "Open" button. Once reloaded into the DEMUSE tool, the results can be freely edited and displayed (graphical representations and animations of the reloaded results are fully supported). "save results" button saves all the decomposition results, including the original surface EMG signals.

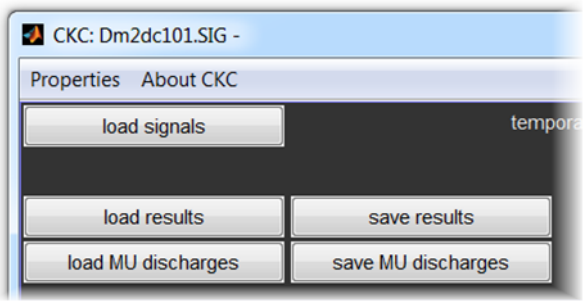


Figure 50: s results" and "load results" buttons.

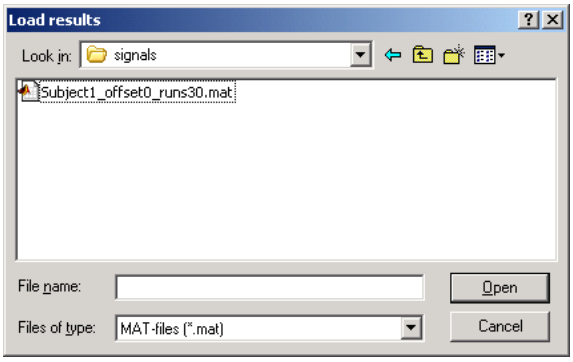
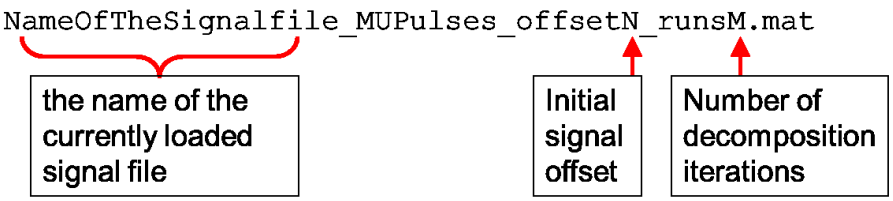


Figure 51: "`Load results`" dialog window.

To save just the discharge patterns of MUs, press the "`save MU discharges`" button (**Figure 50**). MU discharges are automatically saved into the directory containing the currently loaded signal file. The following file naming convention is used:


NameOfTheSignalfile_MUPulses_offsetN_runsM.mat
the name of the currently loaded signal file
Initial signal offset
Number of decomposition iterations

where `NameOfTheSignalfile` stands for the name of the currently loaded signal file, `N` is the initial signal offset (in seconds) and `M` is the number of decomposition runs (see Section 4.5 for details). MU discharges are saved into a Matlab cell structure `MUPulses`, with discharge times (in samples) of induvidual MU in each cell. For example, discharge times of MU 1 are stored in cell `MUPulses{1}`, discharge times of MU 2 in cell `MUPulses{2}` etc.

5 Appendix I: CKCreader

When loading the EMG files, DEMUSE tool prompts for selection of a proper reader. The later can be implemented in Matlab and added to the `DCKCreaders` directory (located in the main directory of the DEMUSE tool). The name of the reader must start with the string "`CKCreader`", but can continue with arbitrary string. A good practice is to specify the main parameters of the reader in its name, so the user can easily identify it. For example, reader

```
CKCreader_5x13_IED5mm_telescopic_pins_SD.m
```

denotes the reader which reads surface EMG acquired by a matrix of 5x13 electrodes with inter-electrode distance of 5 mm and electrodes on telescopic pins. EMG signals were acquired in single-differential (SD) mode.

DEMUSE tool automatically loads all the files whose name starts with "`CKCreader`" into the list of available readers and display their descriptions in the `getCKCReader` Dialog Window (**Figure 7**). Reader for specific files can also be specified in a text file. Simply write the name of the main Matlab file with the reader in the text file called `CKC_reader.txt`, e.g.:

```
CKC_reader_my reader.m
```

and save the file into the directory with corresponding EMG files. DEMUSE tool will automatically check the directory with EMG files for `CKC_reader.txt` file and, if found, use the reader specified therein for all the EMG files in the corresponding directory.

Structure of reader is exemplified in **Figure 52**. All the text before the reserved keyword '`INPUTS`' is considered as a description and displayed in the `getCKCReader` Dialog Window (**Figure 7**).

Inputs to the reader are limited to the path and name of the EMG file and (optionally) the length of the signal to be loaded (in seconds). If no signal length is specified, the reader should return the entire signal in the file.

```
function data = CKC_reader_myreader(filepath,filename,epoch_length)
% function data = CKC_reader_myreader(filepath,filename,epoch_length)
%
% CKC reader: reads surface EMG acquired by a matrix of 5x13
% electrodes with inter-electrode distance of 8 mm.
%
% INPUTS:
%   - filepath: directory with the SIG file to be loaded
%   - finename: SIG file to be loaded
%   - epoch_length: (optional) length of the epoch of signal to be loaded (in s)
%
% OUTPUT:
%   data: structure with the following fields;
%       SIG - two dimensional cell array with surface EMG channel in each
%             cell - SIG{r,c} is the channel in row r and column c. Missing
%             electrodes are denoted by empty arrays, e.g. SIG{1,1} = [];
%       fsamp - sampling frequency of sEMG
%       signal_length - length of a surface EMG signals (in samples)
%       montage - montage of electrodes - 'MONO' for monopolar, 'SD' for
%                 single differential
%       IED - inter-electrode distance (in mm)
%       ref_signal - measured reference signal (e.g., force) if avalable,
%                    empty array otherwise
%       description - arbitrary string describing the data
%
% ------------------------------------------------------------------------
% Author: Ales Holobar (ales.holobar@um.si)
% Last modified: 14. 10. 2012

if nargin < 3
    epoch_length = inf;
end
```

description of a reader (anything before the keyword INPUT will appear in the getCKCReader Dialog Window)

Inputs to the reader are standardized.

Outputs are always in the form of Matlab structure.

Figure 52: Reserved head of a EMG reader and its standardized interface. All the CKCreaders must specify this initial structure and must use specified inputs and outputs. Actual implementation of the loading method is left to the end user.

Outputs of the reader are given in the form of Matlab structure with the following field requested:

- **`SIG`**: two dimensional cell array with surface EMG channel in each cell - `SIG{r,c}` is the channel in the r-th row r and c-th column. Missing electrodes are denoted by empty arrays, e.g. `SIG{1,1} = []`.
- **`fsamp`**: sampling frequency of sEMG.
- **`signal_length`**: length of a surface EMG signals (in samples).
- **`montage`**: montage of electrodes - 'MONO' for monopolar, 'SD' for single differential configuration.
- **`IED`**: inter-electrode distance (in mm).
- **`ref_signal`**: measured reference signal (e.g., exerted muscle force) if available, empty array otherwise.
- **`description`**: arbitrary string describing the data (this string is displayed as the name of main DEMUSE tool window).

Citations and technical support

The development and implementation of DEMUSE tool was supported by the following EU projects:

- **DEMUSE** (Contract No. 023537).
- **iMOVE** (Contract No. 239216).
- **TREMOR** (Contract No. ICT-2007-224051).
- **NeuroTREMOR** (Contract No. ICT-2011.5.1-287739).

The DEMUSE tool should be cited as:

1. A. Holobar, D. Zazula: Gradient Convolution Kernel Compensation Applied to Surface Electromyograms, ICA 2007, LNCS 4666, pp. 617–624, 2007a.

2. A. Holobar and D. Zazula: Multichannel Blind Source Separation Using Convolution Kernel Compensation, IEEE Trans. Sig. Process. 55 (9), 4487-4496, 2007b.

For technical assistance and support, please contact:

Aleš Holobar
System Software Laboratory (SSL)
University of Maribor, Faculty of Electrical Engineering and computer science,
Smetanova ulica 17,
2000 Maribor
Slovenia

phone: +386 2220 7485
e-mail: ales.holobar@um.si
web: storm.uni-mb.si

References

3. A. Holobar, D. Zazula: Gradient Convolution Kernel Compensation Applied to Surface Electromyograms, ICA 2007, LNCS 4666, pp. 617–624, 2007a.

4. A. Holobar and D. Zazula: Multichannel Blind Source Separation Using Convolution Kernel Compensation, IEEE Trans. Sig. Process. 55 (9), 4487-4496, 2007b.

5. A. Holobar, V. Glaser, J.A. Gallego, J.L. Dideriksen, D. Farina: Non-invasive characterization of motor unit behaviour in pathological tremor. *Journal of neural engineering* 2012, doi: 10.1088/1741-2560/9/5/056011.

6. A. Holobar, M. A. Minetto, D. Farina: Accurate identification of motor unit discharge patterns from high-density surface EMG and validation with a novel signal-based performance metric. *Journal of neural engineering*, 2014, doi: 10.1088/1741-2560/11/1/016008.